

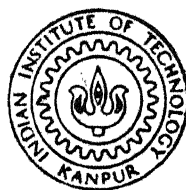
**A FINITE ELEMENT STUDY OF BODY-FITTING AND ADAPTIVE
GRID GENERATION WITH APPLICATION TO FLAME
PROPAGATION PROBLEMS**

By

VUDUTALA CHINA VENKATA RAO

MATH
1992
D
RAO
FIN

TH
MATH/1992/D
R1Bf



**DEPARTMENT OF MATHEMATICS
INDIAN INSTITUTE OF TECHNOLOGY KANPUR
DECEMBER, 1992**

**A FINITE ELEMENT STUDY OF BODY-FITTING AND ADAPTIVE
GRID GENERATION WITH APPLICATION TO FLAME
PROPAGATION PROBLEMS**

*A Thesis Submitted
In Partial Fulfilment of the Requirements
for the Degree of*
DOCTOR OF PHILOSOPHY

By
VUDUTALA CHINA VENKATA RAO

to the
**DEPARTMENT OF MATHEMATICS
INDIAN INSTITUTE OF TECHNOLOGY KANPUR
DECEMBER, 1992**

515.353
R18

- 3 FEB 1994 / Math

CENTRAL LIBRARY
I. I. T., KANPUR

Acc. No. A.1.17.195

MATH-1995-D-RAO-FIN

16.12.92
Sf-kg

C E R T I F I C A T E

It is certified that the work contained in the thesis entitled *A Finite Element Study of Body-fitting and Adaptive Grid Generation with Application to Flame Propagation Problems*, by Vudutala China Venkatarao, has been carried out under our supervision and that this work has not been submitted elsewhere for a degree.

J. Sundararajan
(Dr. T. Sundararajan)

Dept. of Mech. Engg.
Indian Institute of Technology
Kanpur - 208016, India

P. C. Das
(Dr. P. C. Das)

Dept. of Mathematics.
Indian Institute of technology
Kanpur - 208016, India

December, 1992.

SYNOPSIS

Automatic generation of the numerical grid is an important step in the computational simulation of a variety of applied problems. The requirements of automatic grid generation vary from problem to problem. In some cases, the grid is expected to smoothly fit a complicated geometry and yet have grid lines which intersect each other almost orthogonally (Thompson, 1984). For situations which involve a moving front (say, a propagating flame or a shock wave), capturing the location of the front and depicting the solution accurately near the front, become the primary objectives. In order to achieve these objectives, it is essential to generate the numerical grid in an adaptive fashion depending on the computed solution.

In the present study, strategies for automatic grid generation have been developed based on the finite element approach, for both adaptive and non-adaptive situations. The applicability of these techniques has been illustrated on problems having complex geometries or moving fronts.

The non-adaptive grid generation procedure developed in the present study is based on the solution of a Laplacian system of equations. It is meant for the generation of a structured, smooth grid inside an arbitrary shaped region. In this technique, the mapping of the complex physical domain into a simple shaped transformed domain is performed, with the help of body-fitting curvilinear coordinates (ξ, η) (Fig. 1).

$$\left. \begin{aligned} \frac{\partial T}{\partial t} - \frac{\partial^2 T}{\partial x^2} &= f(\rho, T) \\ \frac{\partial \rho}{\partial t} - \frac{\partial^2 T}{\partial x^2} &= -f(\rho, T) \end{aligned} \right\} \quad t > 0, x \in (0, 1) \quad (1-a)$$

with boundary conditions

$$\begin{aligned} \frac{\partial \rho(0, t)}{\partial x} &= 0 = \frac{\partial \rho(1, t)}{\partial x} & t > 0 \\ \frac{\partial T(0, t)}{\partial x} &= 0, \quad T(1, t) = g(t) & t > 0 \end{aligned} \quad (1-b)$$

and initial conditions

$$\rho(x, 0) = 1, \quad T(x, 0) = 0.2, \quad x \in [0, 1] \quad (1-c)$$

Here $f(\rho, T)$ and $g(t)$ represent the following :

$$f(\rho, T) = 3.52 \times 10^6 \rho \exp(-4/T)$$

$$\begin{aligned} \text{and} \quad g(t) &= 0.2 + t/(2 \times 10^{-4}) & t \leq 2 \times 10^{-4} \\ &= 1.2 & t \geq 2 \times 10^{-4} \end{aligned}$$

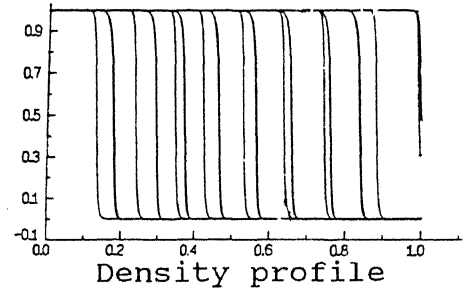
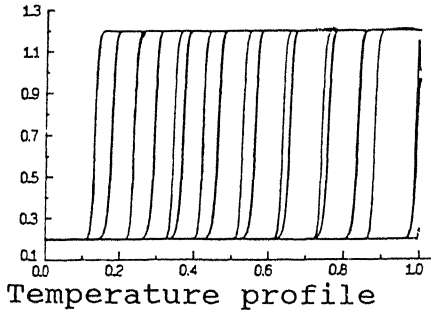


Fig 2 : Propagation of the flame front.

A principle tool for such adaptive mesh generation is the development of an error estimator with a local *a posteriori* character, relating the accuracy of the computed solution to a perturbation of the original problem. An efficient *a posteriori* error estimate for finite element solutions has been derived in an

asymptotic form for step size h tending to zero, by Babuska and Rheinboldt (1978a) for a certain class of elliptic problems where the energy norm of the error is estimated as follows :

$$||e||_E \leq c \left(\sum_{j=0}^{n-1} \eta_j^2 \right)^{1/2} (1+O(h)) \text{ as } h \rightarrow 0 \quad (2)$$

Here computable quantity η_j is called the error indicator and is expressed by :

$$\eta_j^2 = \frac{h_j^2}{12} \int_{x_j}^{x_{j+1}} r_j^2 dx, \quad j = 0, 1, 2, \dots, n-1 \quad (3)$$

where n is the number of nodes in the domain, c is the constant and $h_j = x_{j+1} - x_j$. The *a posteriori* error in the equation (2) provides an easily computable bound for the total solution error under the energy norm. Also, the error indicator η_j provides a measure of error contribution of each element based on the local elemental residue r_j of the element j of the mesh. Using this concept of error analysis, Bieterman and Babuska (1982a) proposed an *a posteriori* error for the time dependent spatial meshes and presented the theory for the estimation of $||e(t, \cdot)||_E$ at each time.

In the present study, the error estimates stated above are used in the h / hp -version adaption strategies. The *a posteriori* error estimate is defined by:

$$\varepsilon(t) = \left(\sum_{j=0}^{n-1} \eta_j^2(t) \right)^{1/2} \text{ for each } t \quad (4)$$

where the computable local error indicator η_j is expressed as :

$$\eta_j^2 = \frac{h_j^2}{12p_j^2} \int_{x_j}^{x_{j+1}} r_j^2 dx, \quad j = 0, 1, 2, \dots, n-1 \quad (5)$$

Here, p_j represents the local degree of finite element approximation considered in the j^{th} element for the h- and hp-methods. The fundamental principle employed for grid adaption is the equi-distribution of the local error indicator η_j within the solution domain, while the *a posteriori* error $\varepsilon(t)$ is brought down below preset tolerance level δ for each time.

In the h-version method, the elements have been refined according to a statistical procedure based on the distribution of local error indicators (Oden, 1986b). The level of refinement or de-refinement is selected on the basis of the deviation of the local error indicator from the mean value of the local error indicators. Such a statistical procedure involving simultaneous refinement and de-refinement at different parts of the solution domain is applied for each time step until the global *a posteriori* error meets the tolerance criterion.

In the h-p method, the p-version and the h-version approaches are used alternately. For the p-strategy, the polynomial degree of interpolation in an element is selected on the basis of the deviation from the mean of the error indicators. During the iterative adaption scheme, the degree of interpolation is incremented (up to a maximum value of five) and the global error criterion is checked for each adaption cycle. If the maximum allowable degree of interpolation is reached without satisfying the global error tolerance, then the concerned highest degree elements are divided according to the h-version strategy. In order to derive maximum benefit of the earlier p-version calculations, a gradient based non-uniform h-refinement has been introduced in the elements of highest degree approximation.

Both the h- and hp-methods have been used to solve a test

problem with front propagation features and having a known analytical solution. Further, and the flame propagation problem given by equation (1) has also been solved. The exact and the computed solutions for the test problem have been compared extensively in respect of the solution profiles, the time-wise variations of quantities such as relative error, effectivity ratio, local and mean error indicators etc., (Bieterman and Babuska, 1982a). The h-version adaption method is seen to be more effective in bringing down the error than the hp-method. However, for the same global error tolerance, the number of elements required by the h-version is considerably higher. An elaborate comparison of the various error measures for the test problem indicates that the *a posteriori* error $\mathcal{E}(t)$ very closely represents the norm of the gradient error.

Apart from the h- and hp-methods, a grid velocity scheme has been implemented in which the grid is moved with the speed of the front after the initial ignition phase. Such a grid speed strategy involves the least amount of computation for grid generation and it is also effective with respect to maintaining the solution error at a low value. But it is restricted to problems which have constant values of front width as well as front speed.

A major contribution of the present work relates to the development of a generalized equi-distribution strategy (designated as the α -strategy) for the local error indicator and a *a posteriori* error. Since the error estimators proposed by Bieterman and Babuska (1982a) are meant for self-adjoint spatial operators, their applicability to other types of problems, especially non-linear problems, is only tentative. The proposed α -strategy circumvents this difficulty. The only assumptions invoked for the

application of α - strategy are that the error estimator depends on the product of measure of the local residue in some properly chosen norm and positive power of the local mesh size. Based on these requirements the local error indicator used for the *a posteriori* error $\varepsilon(t)$ can be expressed as follows:

$$\eta_j^2(t) = h_j^\alpha \int_{x_j}^{x_{j+1}} r_j^2 dx \quad j = 0, 1, 2, \dots, n-1 \quad (6)$$

where the index α is itself an unknown and is to be determined after the calculation of the approximated solution. For any given mesh, the value of α is determined by equating the *a posteriori* error with a prescribed global tolerance limit δ . The degree of refinement for the j^{th} element is decided according to the statistical procedure described earlier. Two limits of tolerance namely, the upper and the lower tolerances δ_1 and δ_2 , are prescribed for the *a posteriori* error. The upper limit δ_1 is used to start the grid adaption, in the sense that whenever $\varepsilon(t) \geq \delta_1$, grid is adapted. The lower limit is similarly used for stopping the adaption as well as for the determination of α .

In order to illustrate the α -strategy, the problem of flame propagation in the presence of convective flow has been solved. This problem is described by the non-self-adjoint operator of the form

$$\begin{aligned} \frac{\partial T}{\partial t} - \frac{\partial^2 T}{\partial x^2} + v \frac{\partial T}{\partial x} &= f(\rho, T) \\ \frac{\partial \rho}{\partial t} - \frac{\partial \rho}{\partial x^2} + v \frac{\partial \rho}{\partial x} &= -f(\rho, T) \end{aligned} \quad (7)$$

where v is the fluid velocity. The boundary and initial conditions are prescribed as in equations (1-b) and (1-c) respectively.

The Galerkin approach involving upwind weighting function has

been employed to obtain the finite element space discretization of the above equation, while the Crank-Nicolson scheme is employed for the purpose of time marching. The computed results have been compared with the analytical results of a test problem which displays the front movement feature. It is observed that the α -strategy performs exceedingly well in capturing the front movement and the spatial distribution of the local error indicator moves with front. Further, the computed error estimator approaches the norm of the actual gradient error, after a few initial cycles of grid adaption. For refined meshes, the value of α approaches two for a non-convective problem ($V = 0$) which agrees with the error estimate proposed by Bieterman and Babuska (1982a,b). The behavior of all the error measures indicates that the α -strategy is very effective in estimating and equi-distributing the solution error, in a self corrective fashion.

ACKNOWLEDGEMENT

I wish to express a deep sense of gratitude to my advisors Dr. P. C. Das and Dr. T. Sundararajan for their guidance, support and alround consistent encouragement throughout the course of this research. I feel very fortunate to have the opportunity of getting them as my supervisors. I will always cherish my days of working with them as a potential learning experience and shall always remain obliged to their greatness in devoting such a large share of their valuable time and knowledge to help me complete my work.

I would gratefully recall the favour and encouragement I received from Dr. M. R. M. Rao, Dr. M. K. Kadalbajoo, Dr. K. Muralidhar and Dr. G. Biswas during my programme

I would like to thank my colleagues at IIT Kanpur especially Dr. Saha, Dr. Mukho, Dr. Moustafa, Sudhakar, Tara, Mandal, Prasad and all other friends for their kind help during the final phase of my thesis work.

I also express my sincere gratitude to my family members for showing considerable patience during my long absence from my home. Without their cooperation, the present work would have been an impossibility.

CONTENTS

| | <u>Page</u> |
|---|-------------|
| Title Page | i |
| Certificate | ii |
| Synopsis | iii |
| Acknowledgement | xi |
| Contents | xii |
| List of Figures | xvi |
| List of Tables | xxii |
| Nomenclature | xxiii |
| CHAPTER 1. | |
| INTRODUCTION | 1 |
| 1.1. Motivation | 1 |
| 1.2. Survey of literature | 3 |
| 1.3. Objectives of the Present Work | 9 |
| 1.4. Outline of the Dissertation | 10 |
| CHAPTER 2. | |
| GENERATION OF BODY-FITTING GRIDS USING FINITE ELEMENT METHOD | 12 |
| 2.1. Introduction | 12 |
| 2.2. Literature Survey | 12 |
| 2.2.1. Unstructured grid generation | 14 |
| 2.2.2. Structured grid generation | 14 |
| 2.2.3. Grid generation equations based on elliptic equations | 18 |
| 2.3. Scope of the present work : | 21 |
| 2.4. Geometry of the problems considered | 22 |
| 2.5. Finite element formulation of grid generation equations | 24 |
| 2.6. Description of the algorithm | 29 |
| 2.7. Important features of the algorithm | 32 |
| 2.8. Results and discussion | 34 |
| 2.9. Conclusions | 43 |

CHAPTER 3.

| | |
|---|----|
| FINITE ELEMENT FORMULATION OF FLAME PROPAGATION PROBLEMS AND DESCRIPTION OF COMPUTABLE ERROR ESTIMATORS FOR GRID ADAPTION | 44 |
| 3.1. Introduction | 44 |
| 3.2. Modelling of flame propagation problem | |
| 3.2.1. General background | 45 |
| 3.2.2. Formulation of flame propagation problem | 48 |
| 3.2.3. Salient features of the flame propagation problem | 53 |
| 3.2.4. Computational requirements of flame propagation studies | 54 |
| 3.3. Mathematical formulation and finite element solution procedure | 55 |
| 3.3.1. Mathematical preliminaries | 56 |
| 3.3.2. Basic formulation in a general set up | 58 |
| 3.3.3. Method of solution | 63 |
| 3.4. Description of error measures used for grid adaption in finite element method | 68 |
| 3.4.1. History of a posteriori error estimates used for grid adaption | 69 |
| 3.4.2. Description of a posteriori error estimates | 71 |
| 3.4.3. A posteriori error estimates used in present work | 77 |

CHAPTER 4.

| | |
|--|----|
| ADAPTIVE GRID TECHNIQUES FOR FLAME PROPAGATION PROBLEM IN A STATIONARY MIXTURE | 78 |
| 4.1. Introduction | 78 |
| 4.2. History of error based adaption strategies | 78 |
| 4.2.1. Adaptive strategies by Finite Difference Methods | 79 |
| 4.2.2. Adaptive strategies by Finite Element Methods | 80 |
| 4.3. Problem Description | 83 |
| 4.3.1. Description of test problem | 84 |

| | <u>Page</u> |
|--|-------------|
| 4.3.2. Description of Flame propagation problem | 85 |
| 4.3.3. Adaptive grid Notation | 87 |
| 4.3.4. Finite element formulation | 92 |
| 4.4. Scope of the present study | 93 |
| 4.4.1. Objectives | 93 |
| 4.4.2. Mathematical formulation of the objectives | 94 |
| 4.5. h-version adaptive strategy | 97 |
| 4.5.1. Description of the algorithm | 97 |
| 4.5.2. Important features of the algorithm | 100 |
| 4.5.3. Computational procedure | 110 |
| 4.6. h-p version adaptive strategy | 113 |
| 4.6.1. Description of the algorithm | 113 |
| 4.6.2. Important features of the algorithm | 118 |
| 4.6.3. Computational Procedure | 134 |
| 4.7. Grid speed adaptive strategy | 137 |
| 4.7.1. Description of Grid speed technique | 137 |
| 4.7.2. Description of the algorithm | 138 |
| 4.7.3. Computational Procedure | 141 |
| 4.8. Results and discussion | 142 |
| 4.8.1. Application of adaptive strategies to Test problem | 142 |
| 4.8.2. Application of adaptive strategies to the Flame propagation problem | 173 |
| 4.8.3. Computational time | 220 |
| 4.9. Conclusions | 227 |

CHAPTER 5.

| | |
|--|-----|
| A SELF ADAPTIVE ERROR INDEX BASED REFINEMENT STRATEGY APPLIED TO FLAME PROPAGATION IN PRESENCE OF FLOW | 228 |
| 5.1. Introduction | 228 |
| 5.2. Previous work on convective problems : Adaptive strategies | 228 |
| 5.3. Description of the present work | 231 |
| 5.3.1. Description of Test problem | 232 |

| | <u>Page</u> |
|---|-------------|
| 5.3.2. Description of Flame problem in presence of flow | 233 |
| 5.3.3. Basic formulation | 233 |
| 5.4. Description of present a posteriori error estimate | 239 |
| 5.5. Scope of the present study | 240 |
| 5.5.1. Mathematical statement | 241 |
| 5.6. h version Adaptive strategy | 244 |
| 5.6.1. Description of algorithm | 244 |
| 5.6.2. Important features of algorithm | 249 |
| 5.7. Results and discussions | 252 |
| 5.7.1. Application of α -strategy to Test problem | 253 |
| 5.7.2. Application of α -strategy to Flame problem | 273 |
| 5.7.3. Computational time | 303 |
| 5.8. Conclusions | 305 |
| CHAPTER 6. | |
| CONCLUSIONS AND SUGGESTIONS FOR FUTUER WORK | 306 |
| 6.1. Conclusions of present work | 306 |
| 6.2. Suggestions for future work | 308 |
| REFERENCES. | 310 |
| ADDENDUM | 328 |

LIST OF FIGURES

| Fig. | TITLE | Page |
|------|--|------|
| 2.1a | Transformation of an ellipse to a rectangle | 19 |
| 2.1b | Transformation of an annular region to a rectangle | 19 |
| 2.2 | Eight noded isoparametric element and the local coordinate system | 25 |
| 2.3 | Construction of ξ -constant lines | 25 |
| 2.4 | Flow chart for the generation of grid | 31 |
| 2.5 | Numerical grid for the ellipse domain | 36 |
| 2.6 | Numerical grid for the simply connected domain : Effect of uniform grid spacing | 37 |
| 2.7 | Numerical grid for the simply connected domain : Effect of dense boundary data | 38 |
| 2.8 | Numerical grid for the elliptic domain : Effect of non-smooth boundaries | 40 |
| 2.9 | O-type grid for dense boundary data around concave corners | 41 |
| 2.10 | O-type grid for uniformly distributed boundary data around | 42 |
| 2.11 | O-type grid for arbitrary shaped branch cut with a very bad initial guess | 42 |
| 3.1 | A schematic diagram of one dimensional flame propagation | 49 |
| 3.2 | Schematic diagram of flame propagation inside spark ignition engine | 51 |
| 4.1 | Test problem solution profile at different times | 86 |
| 4.2 | Flow chart for h version algorithm | 99 |
| 4.3 | Schematic representation of the evolution of a posteriori error with adaptations / time in h-version algorithm | 103 |
| 4.4 | Flow chart for h-p version algorithm | 115 |
| 4.5 | Hierarchical element shape functions of nearly orthogonal form | 119 |

| Fig. | TITLE | Page |
|---------|--|------|
| 4.6a | Schematic representation of the evolution of a posteriori error with adaptations / time in h-p version algorithm | 123 |
| 4.6b | Approximation of computed FE solution during p to h conversion in a higher order element | 131 |
| 4.7 | Flow chart for grid speed strategy | 140 |
| 4.8a,b | Test problem (uniform mesh): Solution profile ((a) $N = 20$, (b) $N = 40$) | 145 |
| 4.8c,d | Test problem (uniform mesh): Solution profile ((c) $N = 80$, (d) $N = 160$) | 146 |
| 4.9 | Test problem: Time evolution of errors (uniform mesh) (A posteriori error and true gradient error) | 151 |
| 4.10 | Test problem: Time evolution of errors (uniform mesh) (relative error and effectivity ratio) | 152 |
| 4.11a,b | Test problem (h and hp version): Solution profile | 155 |
| 4.12 | Test problem: Time evolution of errors (h and hp version) | 158 |
| 4.13 | Test problem: Time evolution of errors (h and hp version) maximum and mean error indicators | 160 |
| 4.14 | Test problem (h and hp version): Adapted mesh at various t | 164 |
| 4.15 | Test Problem: Time evolution of errors (h version) (Non-smooth and smooth effect) | 168 |
| 4.16 | Test problem (grid speed): Solution profile | 171 |
| 4.17 | Test problem (grid speed): Adapted mesh at various t | 171 |
| 4.18 | Test problem: Time evolution of errors (Grid speed) | 172 |
| 4.19 | Flame problem (uniform mesh): Solution profile ($N = 40, N = 80$) | 175 |
| 4.20 | Flame problem (uniform mesh): Solution profile ($N = 160, N = 320$) | 176 |
| 4.21 | Flame problem: Time evolution of errors (uniform mesh) (A posteriori error and gradient) | 180 |

| Fig. | TITLE | Page |
|-------|--|------|
| 4.22 | Flame problem (h and uniform mesh): Solution profile | 183 |
| 4.23 | Flame problem (h and hp version): Solution profile | 184 |
| 4.24 | Flame problem: Time evolution of errors (h and hp version) (A posteriori error and gradient) | 185 |
| 4.25 | Flame problem: Time evolution of errors (h and hp version) (Maximum and mean error indicators) | 191 |
| 4.26a | Flame problem: Behaviour of errors (h version) ((i) $t = 0.00001$, (ii) $t = 0.00003$) | 194 |
| 4.26b | Flame problem: Behaviour of errors (h version) ((i) $t=0.000145$, (ii) $t=0.00030$, (iii) $t=0.00070$) | 195 |
| 4.26c | Flame problem: Behaviour of errors (h version) ((i) $t = 0.00071$, (ii) $t = 0.00105$, (iii) $t = 0.00155$) | 196 |
| 4.26d | Flame problem: Behaviour of errors (h version) ((i) $t = 0.002025$, (ii) $t = 0.002565$, (iii) $t = 0.003025$) | 197 |
| 4.26e | Flame problem: Behaviour of errors (h version) ((i) $t = 0.003570$, (ii) $t = 0.003995$, (iii) $t = 0.004545$) | 198 |
| 4.26f | Flame problem: Behaviour of errors (h version) ((i) $t = 0.004995$, (ii) $t = 0.005580$, (iii) $t = 0.006035$) | 199 |
| 4.27a | h-Adapted mesh at various time instants (Flame problem) | 201 |
| 4.27b | h-Adapted mesh at various time instants (Flame problem) | 202 |
| 4.28a | Flame problem: Behaviour of errors (hp version) ($t = 0.00001$) | 204 |
| 4.28b | Flame problem: Behaviour of errors (hp version) ((i) $t = 0.00002$, (ii) $t = 0.00003$) | 205 |
| 4.28c | Flame problem: Behaviour of errors (hp version) ($t = 0.00011$) | 206 |
| 4.28d | Flame problem: Behaviour of errors (hp version) ((i) $t = 0.000145$, (ii) $t = 0.0003$) | 207 |

| Fig. | TITLE | |
|--------|--|-----|
| 4.28e | Flame problem: Behaviour of errors (hp version) ((i) $t = 0.00054$, (ii) $t = 0.001015$, (iii) $t = 0.002305$) | 208 |
| 4.28f | Flame problem: Behaviour of errors (hp version) ((i) $t = 0.00306$, (ii) $t = 0.004310$, (iii) $t = 0.005695$) | 209 |
| 4.29a | hp - Adapted mesh at various time instants | 214 |
| 4.29b | hp - Adapted mesh at various time instants | 215 |
| 4.29c | hp - Adapted mesh at various time instants | 216 |
| 4.30 | Flame problem (h and hp version): Adapted mesh at various t | 217 |
| 4.31a | Flame problem (h version & Grid speed): Solution profile | 218 |
| 4.31b | Flame problem (Grid speed): Adapted mesh at various t | 218 |
| 4.32 | Flame problem: Time evolution of errors (Grid speed) | 221 |
| 5.1 | Construction of quadratically biased upwind test functions | 236 |
| 5.2 | Flow chart for h-version based α strategy | 245 |
| 5.3 | Schematic representation of the evolution of a posteriori error with adaptations/time in h version based α strategy | 251 |
| 5.4a,b | Test problem (h version) : Solution Profile | 255 |
| 5.4c | Test problem (h version) : Solution Profile | 256 |
| 5.5a | Test problem (h version) : Adapted mesh at various t | 256 |
| 5.5b,c | Test problem (h version) : Adapted mesh at various t | 257 |
| 5.6a | Test problem : Adapted mesh at various times ($V = 5.0$) | 259 |
| 5.6b | Test problem : Adapted mesh at various times ($V = 5.0$) | 260 |
| 5.7 | Test problem : Time evolution of alfa values (h version) | 262 |

xx

| Fig. | TITLE | Page |
|------|---|------|
| 5.8 | Test problem : Time evolution of errors (h version) (A posteriori error and gradient error) | 265 |
| 5.9 | Test problem : Time evolution of errors (h version) (Maximum and Mean error indicators) | 266 |
| 5.10 | Test problem : Time evolution of residues (h version) (Maximum and Mean residue indicators) | 268 |
| 5.11 | Test problem : Time evolution of residue and global gradient (h version) | 269 |
| 5.12 | Test problem : Behaviour of errors (h version : V = 5.0) (t = 0.00005) | 270 |
| 5.13 | Test problem : Behaviour of errors (h version : V = 5.0) ((i) t = 0.0123, (ii) t = 0.02775, (iii) t = 0.03560) | 271 |
| 5.14 | Test problem : Behaviour of errors (h version : V = 5.0) ((i) t = 0.0514, (ii) t = 0.0687 , (iii) t = 0.07885) | 272 |
| 5.15 | Test problem (h version) : Solution profile (V = 0.0) | 274 |
| 5.16 | Test profile (h version) : Adapted mesh at various t(V = 0.0) | 274 |
| 5.17 | Test problem : Behaviour of errors (h version : V= 0.0) (t = 0.00005) | 275 |
| 5.18 | Test problem : Behaviour of errors (h version : V= 0.0) ((i) t = 0.01045, (ii) t = 0.01615 (iii) t = 0.02725) | 276 |
| 5.19 | Test problem : Behaviour of errors (h version : V= 0.0) ((i) t = 0.03955, (ii) t = 0.05915 (iii) t = 0.07995) | 277 |
| 5.20 | Test problem : Time evolution of alfa values, residues and errors (h version : V=0.0) | 278 |
| 5.21 | Flame problem (Uniform and h version mesh : V = 5.0) : Solution profile | 281 |
| 5.22 | Flame problem (h version) : Solution profile (V = 200.0) | 283 |
| 5.23 | Flame problem (h version) : Solution profile (V = 0.0) | 285 |
| 5.24 | Flame profile (h version) : Adapted mesh at various t (V = 0.0) | 285 |

| Fig. | TITLE | Page |
|-------|---|------|
| 5.25 | Flame problem (h version) adapted mesh at various t (V = 5.0) | 286 |
| 5.26a | Flame problem :Adapted mesh at various time instants (V = 5.0) | 289 |
| 5.26b | Flame problem :Adapted mesh at various time instants (V = 5.0) | 290 |
| 5.27 | Flame problem : Time evolution of errors and alfa values, (uniform mesh (a) N = 400, (b) N = 200) | 291 |
| 5.28 | Flame problem : Time evolution of alfa values, gradient and errors (h version : V = 5.0) | 293 |
| 5.29 | Flame problem : Time evolution of alfa values, gradient and errors (h version : V = 200.0) | 294 |
| 5.30 | Flame problem : Time evolution of alfa values, gradient and errors (h version : V = 0.0) | 295 |
| 5.31 | Flame problem : Behaviour of errors (h version : V = 0.0) (t = 0.00001) | 297 |
| 5.32 | Flame problem : Behaviour of errors (h version : V = 0.0) ((i) t = 0.00013, (ii) t = 0.00071, (iii) t = 0.00119) | 298 |
| 5.33 | Flame problem : Behaviour of errors (h version : V = 0.0) ((i) t = 0.00205, (ii) t = 0.00266, (iii) t = 0.00318) | 299 |
| 5.34 | Flame problem : Behaviour of errors (h version : V = 5.0) (t = 0.00001) | 300 |
| 5.35 | Flame problem : Behaviour of errors (h version : V = 5.0) ((i) t = 0.00004, (ii) t = 0.00092, (iii) t = 0.00180) | 301 |
| 5.36 | Flame problem : Behaviour of errors (h version : V = 5.0) ((i) t = 0.00468, (ii) t = 0.00527, (iii) t = 0.00600) | 302 |

LIST OF TABLES

| No. | TITLE | Page |
|------|---|------|
| 4.1 | Schematic representation of variables used in the h and h-p version algorithm | 91 |
| 4.2a | Selection of grid points in an element of p to h conversions | 129 |
| 4.2b | Notations for variables used in the subsequent tables | 144 |
| 4.3a | A posteriori error predictions | 147 |
| 4.3b | Gradient of true error predictions | 147 |
| 4.3c | Relative error predictions | 148 |
| 4.3d | Effectivity ratio predictions | 148 |
| 4.4a | Test problem results for h-version strategy | 162 |
| 4.4b | Test problem results for h-p version strategy | 163 |
| 4.5 | h-version results for flame propagation problem | 187 |
| 4.6a | h-p version results for flame propagation problem | 188 |
| 4.6b | h-p version results for flame propagation problem | 189 |
| 4.7 | Results of the grid speed strategy for the flame propagation problem | 222 |
| 4.8 | CPU time of the h version algorithm for the flame propagation problem | 223 |
| 4.9 | CPU time of the grid speed strategy for the flame propagation problem | 223 |
| 4.10 | CPU time of the h-p version algorithm for the flame propagation problem | 224 |
| 5.1 | Test problem results for h version based α strategy ($V = 0.0$) | 279 |
| 5.2 | Flame propagation results for h version based α strategy ($V = 0.0$) | 287 |
| 5.3 | CPU time of the h version based α strategy algorithm for the flame propagation ($V = 0.0$) | 304 |

NOMENCLATURE

| | |
|--------------------------------|---|
| $[c]$ | matrix of size $(N_h \times N_h)$ |
| $[\tilde{c}]$ | upwinding matrix of size $(N_h \times N_h)$ |
| $\{d\}$ | vector of dimension N_h |
| $\varepsilon^{1,m}(t)$ | a posteriori error at $t = t_1$ for m^{th} adaption. |
| $\text{Deg}(\Omega_n)$ | degree of approximation used in an element |
| $E_n^{1,m}$ | elemental (n) error for all equations |
| h_j | length of the element Ω_j |
| $N(\Delta)$ | dimension of finite dimensional solution space |
| $N(\xi)$ | number of grid points in the η -curve. |
| $N(\eta)$ | number of grid points in the ξ -curve. |
| NPDE | number of partial differential equations. |
| N_h | dimension of the solution space. |
| \mathcal{P} | family of partitions |
| $\mathcal{R}^{1,m}(t, \alpha)$ | residue of the solution |
| S_h | finite dimensional solution space. |
| $S^{r,k}$ | finite dimensional solution space |
| $\tilde{u}_1(x, t)$ | weak solution |
| $U_1(x, t)$ | continuous time Galerkin approximation |
| $U_1[t]$ | solution vector |
| $\theta_1^{1,m}(x, t)$ | approximation for the initial condition |
| $\hat{U}_1^{1,m}(x, t)$ | starting guess solution |
| $\bar{U}_1^{1,m}(x, t)$ | computed FE solution |
| V, \mathcal{V} | prescribed velocity |
| w | upwind parameter |
| $\alpha(x)$ | basis function in test solution space |
| (x, y) | coordinates in the physical plane |

local coordiantes used in each element

power of mesh size of an element

additional unknowns for hierarchical variables

lower tolerance parameter

upper tolerance parameter

boundary of the domain

mesh partition

body-fitting coordinates in the transformed plane

elemental (n) error indicator

approximation to the domain Ω

element j of the mesh Ω .

solution domain

basis function for the node i

basis function for the node j

upwind biased basis functions

corresponding to a typical element

finite dimensional approximation

indices denoting node or element number

i^{th} partial differential equation

n^{th} element

computed solution

starting guess solution

| | |
|-------------|---|
| \sim | approximation in the weak formulation (upwind matrix contribution) |
| o | initial condition |
| i | at time t_i |
| \tilde{k} | vector used in test space of upwind formulation |
| m | m^{th} adaption |
| M | final adaption |
| \bar{p} | vector used in solution space |

CHAPTER 1

INTRODUCTION

1.1. Motivation :

Numerical solution of engineering problems has made great strides in recent years. Diverse fields such as aerodynamics, structural mechanics, process metallurgy etc. employ numerical simulation for analysing the optimality of a design or the efficiency of a process. A significant number of the application problems in these fields involve the solution of coupled partial differential equations in complex geometries. Over the past three decades, several numerical techniques have been developed for the solution of such partial differential equation (PDE) systems in regular domains. The focus of recent research is in the development of numerical techniques and software which facilitate the handling of complex geometries. A large volume of research literature has appeared on the generation of numerical grids in arbitrary- shaped geometries and various techniques have been proposed for generating meshes which are suitable for the application of finite difference (FD), finite volume (FV) and finite element (FE) methods in solving the coupled PDE systems. A field of special interest is the dynamic generation of numerical grids for problems involving the propagation of sharp fronts. For instance, in premixed flame combustion, a flame front may propagate; in transient supersonic flows, moving shock fronts may occur. For the accurate capturing of such sharp fronts and for obtaining the overall solution to these problems, one needs to

take recourse to dynamic grid generation algorithms.

For the development of grids in complex geometries, the use of body- fitting coordinate systems has found wide appeal in computational fluid dynamics and related fields. In order to resolve sharp fronts and the associated non-linear phenomena, solution- adaptive mesh generation techniques have been developed. During the generation of numerical grid by the adaptive or non- adaptive methods, several requirements are placed on the evolving grid structure. For instance, the grid lines should fit the geometry well at the boundary and vary smoothly within the interior of the domain. The grid should not become skewed and should not have too many or too few points at any particular region of the domain. Further, the grid spacing should be selected in such a way as to capture sharp fronts in the predicted solution. The grid spacing control is important not only for reducing the numerical error in the solution but also from the point of view of analysing important physical phenomena. Clearly, some of these requirements are mutually conflicting and one needs to find a compromise between them by judicial choice. Depending on the problem at hand and the numerical technique chosen to solve the PDE's, a particular grid characteristic is given more importance than the others.

Although the developments of body- fitting grids for complex geometries and adaptive methods for problems with sharp fronts have advanced at a rapid pace in recent years, there are still several questions which are unanswered. Robust and quickly converging algorithms for the generation of smooth body- fitting grids with desired characteristics, are not yet available.

Further, the application of error- based adaptive strategies to non-linear problems such as flame propagation requires further study. In the present work, some finite element based non-adaptive and adaptive algorithms are developed and applied to arbitrary shaped geometries and flame propagation studies respectively. Detailed analysis of a posteriori error behaviour and the comparative features of alternative algorithms are evaluated.

1.2. Survey of literature:

Comprehensive surveys of literature related to the problems discussed in each of the subsequent chapters, have been presented in the respective chapters. Therefore, only an overview of the research carried out on adaptive and non- adaptive grid generation techniques is presented here.

The grid generation techniques available at present can be classified into two major categories, namely: (i) Structured grid generation, (ii) Unstructured grid generation. Structured grid generation involves the transformation of a complex physical domain into a simple- shaped computational domain (usually rectangular) using curvilinear, body- fitting coordinates. The curvilinear coordinates are derived by one of the methods such as conformal mapping, algebraic interpolation or the solution of suitable elliptic partial differential equations. Structured grids are primarily useful for obtaining FD or FV based solutions. On the other hand, the unstructured grid generation techniques have been mainly used with finite element solution of the physical problem. These methods involve the algorithmic division of the

geometry into elements of chosen shape. Error analyses of the predicted FE solutions are often used during unstructured mesh generation for obtaining optimal mesh distributions in adaptive applications. Several excellent surveys, books and conference proceedings are available which summarize the research done on structured and unstructured grid generation (Anderson, 1983b; Arcilla, 1991; Babuska and Rheinboldt, 1982; Babuska et al., 1983; Eiseman, 1986; Eiseman and Eriebacher, 1987; Ghia and Ghia, 1983; Hawken, 1987; Ho- Le, 1988; Sengupta et al., 1988; Thompson et al., 1982, 1985a; Thompson, 1982a, 1984, 1985). A few important works on the different aspects of grid generation are described below.

(a) Non- adaptive Methods for Structured Body- fitting Grids :

In these methods, the complex geometry is transformed into a simple rectangular domain with the help of body- fitting coordinates (say, ξ and η). The grid is formed by the intersection of the ξ - and η - constant curves. All the computations are performed in the transformed domain and the computational grid remains fixed regardless of the shape or movement of the physical boundaries. The transformation can be achieved analytically through conformal mapping (O' Brien, 1981; Thompson et al., 1985a). However, conformal mapping is possible for regular geometries and two dimensional situations only. Numerical transformations can be obtained by algebraic (interpolation) techniques or through the solutions of elliptic partial differential equations. Algebraic methods are the fastest among all the grid generation techniques. They can be applied to

two- or three- dimensional problems (Smith, 1983; Smith and Eriksson, 1980; Zhu et al., 1988). However, boundary discontinuities are transmitted to the interior by these methods and therefore, obtaining smooth grids is difficult. The most general procedure for generating body- fitting grids involves the solution of elliptic PDE's for the variation of curvilinear coordinates (Eiseman, 1985a; Ghia and Ghia, 1983; Mastin and Thompson, 1978; Thompson et al., 1977; Thompson, 1982b). Smoothness and boundary orthogonality can be achieved in the grids generated by these methods. Indeed, with some compromise on smoothness and orthogonality, good control over grid spacing can also be obtained by selecting appropriate control functions in Poisson- type grid generation equations (Steger and Sorenson, 1979; Sorenson and Steger, 1980). The PDE grid solvers can be associated with variational functionals for grid smoothness, orthogonality and spacing control (Thompson et al., 1985a). The variational approach of obtaining structured grids has been discussed by Saltzman and Brackbill (1982).

(b) Non- adaptive Methods for Unstructured Grids :

Unstructured meshes are generated by algorithmic division of the given geometry into elements of desired shape. These are associated with finite element solution methods. The available unstructured mesh generation techniques have been summarized by Thacker (1980) and Ho- Le (1988). Generation of triangular elements have been discussed by Ho- Le (1988) and Joe & Simpson (1986). Methods for obtaining quadrilateral elements in arbitrary 2-D geometries have been described by Blacker and Stephenson (1991), Talbert and Parkinson (1990), Zhu et al. (1991) and

Zienkiewicz and Phillips (1971). Quad-tree and Octree approaches have been developed which approximate the geometry in terms of step-like figures consisting of squares in 2-D and cubes in 3-D respectively (Kela et al., 1986; Yerry and Shepherd, 1984, 1988). Cavendish et al. (1984) have described another approach to three dimensional mesh generation. Geometrical methods of generating unstructured meshes have been developed by Bachmann et al. (1987), George et al. (1992) and Lo (1991). A versatile two dimensional mesh generator with automatic band width reduction has been proposed by Liu and Chen (1989).

(c) Adaptive Techniques for Structured Grid Generation :

It was described earlier that for elliptic grid solvers, spacing control is possible by proper choice of control functions (Sorenson and Steger, 1980; Steger and Sorenson, 1979; Brackbill, 1982). In adaptive methods, it is aimed to relate the spacing to some measure of local error and determine the spacing so that the spatial error is equi-distributed. The equi-distribution principle can be stated as

$$\int_{x_i}^{x_{i+1}} w(x) dx = \text{constant}$$

or, in discrete form, $h_i w_i = \text{constant},$

where $h_i (= x_{i+1} - x_i)$ is the step size and $w(x)$ is some positive weight function related to the magnitude of the local error in the predicted solution. Several adaptive grid methods have been devised which employ the equi-distribution concept and these are summarized in the review articles by Anderson (1983b), Berger (1987), Eiseman (1986), Ghia et al. (1983), Thompson (1985), Thompson et al. (1985b).

Grid adaption based on the equi-distribution of solution gradients was studied by Dwyer et al. (1980, 1982), Dwyer and Sanders (1983) and Dwyer (1983). Nakahashi and Deiwert (1985) proposed a spring analogy approach to move the grid- points. The grid points were considered to be connected by hypothetical tension and torsion springs. By relating the spring constants to the solution gradients (or any other suitable measure of error), the authors showed that the grid spacing and skewness could be effectively controlled. Saltzman and Brackbill (1982) and Kreis et al. (1986) have described the use of variational methods for adaptive grids.

Movement of grid points with variable speeds until the equi-distribution principle is satisfied, has been suggested by many authors (Rai and Anderson, 1981; Anderson and Rai, 1982; Anderson, 1983a; Nakumara, 1982; Mack et al., 1992). Rai and Anderson (1981) developed the attraction- repulsion strategy for evaluating the grid speed. The nodal speeds were determined on the basis of local truncation error. Nakumara (1982) proposed the use of parabolic partial differential equations for moving the grid points.

(d) Adaptive Methods for Unstructured Grids :

The research literature in this area has been summarized by Babuska and Rheinboldt (1982), Babuska et al. (1983), Carey and Humphrey (1981), Carey and Oden (1984a), Thompson (1985) and Zhu and Zienkiewicz (1991).

Adaptive triangulation procedures have been described by Eiseman (1985b), Eriebacher and Eiseman (1987) and Rivara (1984).

Shepherd et al. (1986) and Yerry and Shepherd (1984) have developed octree based adaptive mesh refinement. Hierarchical FE refinement has been discussed by Adjedj and Aubey (1989) and Yuen et al. (1991). Moving finite elements have been studied by Adjerd and Flaherty (1986a,b), Arney et al. (1986), Gelinass et al. (1981) and Miller and Miller (1981a,b). Adaptive time discretization has been considered by Babuska and Luskin (1989) and the space- time approach has been implemented by Bjer (1989).

The most significant development in adaptive FE procedures is the construction of a posteriori estimates for the spatial error in the computed solution and grid adaption based on the equidistribution of local errors. The research in this area has been pioneered by Babuska and Rheinboldt (1978a-b, 1979a-c, 1980, 1981, 1982), Bank & Weiser (1985), Bank (1986), Bieterman and Babuska (1982a-b, 1986), Carey (1988), Demkowicz et al. (1989), Flaherty et al. (1983), Gago et al. (1983) and Kelly et al (1983). Based on these estimates, h- version methods which refine the grid spacing and p- version techniques which increment the order of FE interpolation have been developed by Babuska and Dorr (1981), Babuska et al. (1981), Babuska et al. (1984), Demkowicz et al. (1989), Gui and Babuska (1986a-c) and Rachowicz et al. (1989). The concept of feedback for grid adaption has been introduced by Rheinboldt (1983), Babuska (1986) and Babuska and Gui (1986).

(e) Applications of Adaptive Grids :

The application which is of primary interest to the present study is the numerical modelling of combustion and the associated flame propagation phenomena. Several studies have been carried out in recent years on this subject. These have been pioneered by

Benkhaldoun and Larrouturou (1987, 1989), Benkhaldoun et al. (1988a-b), Dervieux et al. (1989), Dervieux and Larrouturou (1990), Dwyer and Sanders (1983), Glowinski (1985), Larrouturou (1985, 1986) and Ramos (1983, 1985, 1987, 1990a-b, 1991). Various aspects of combustion and flame propagation have been analysed by the above authors. A majority of these studies employ finite difference based grid adaption or the moving mesh methods. Incorporation of h-version FE adaption has also been carried out by some authors.

1.3 Objectives of the Present Work :

One of the objectives of the present study is to simplify the solution procedure for the elliptic equations used in the generation of structured grids in complex geometries. Although the elliptic PDE based grid generation systems are versatile, their applicability at present is being limited by the strong non-linearity of the transformed equations in the computational domain. Here, it is proposed to solve the elliptic PDE's in the physical domain itself using the finite element method, without transforming them into the rectangular computational domain. By such simplification, a robust and versatile grid generation algorithm can be constructed for obtaining structured grids with the desired level of smoothness, orthogonality and spacing control.

Another major objective is to model one-dimensional flame propagation using the error estimates derived by Babuska and Rheinboldt (1982) in the h- and h-p version refinement algorithms. It is intended to compare the relative performances of the two

methods among themselves and also against the moving mesh methods, for the flame propagation problem . A detailed study of the error behaviour and the optimality of grid distribution, will also be carried out.

The final objective is to develop a generalized self-correcting error estimator strategy which eliminates the need for a detailed mathematical analysis for constructing appropriate a posteriori error estimates. It is intended to test this algorithm on the flame propagation problem which includes fluid flow.

1.4 Outline of the Dissertation :

In the Chapter 2, structured grid generation using the finite element method is described. The method has been applied for the generation of body- fitting grids in simply and multiply connected domains. The effects of factors such as the nature of boundary data specified and the initial guess grid, on the characteristics of the converged grid are examined.

In Chapter 3, the one- dimensional flame propagation during premixed combustion is described. The mathematical framework for the grid adaption study and the definition of a posteriori error estimates and associated quantities, are provided. The finite element solution procedure for the flame propagation in a stationary medium is discussed.

In Chapter 4, the h - , h - p , and grid speed strategies are described. All the steps involved in the grid refinement or the derefinement procedures are elaborately discussed. The predicted results for the three adaption strategies are compared for both the test and the flame problem. The computed results are also

validated with the help of a test problem which has similar front movement features.

In Chapter 5, the generalized self- correcting error estimator strategy is described. Its application is illustrated with the help of the test problem and the flame problem, for a flowing mixture.

In Chapter 6, the main conclusions of the work are summarized and some suggestions for future work are provided.

CHAPTER 2

GENERATION OF BODY-FITTING GRIDS USING FINITE ELEMENT METHOD

2.1. Introduction :

In this chapter, a novel procedure for the automatic numerical generation of a structured, smooth grid with body-fitting coordinates inside an arbitrarily shaped region is discussed. The method is based on Laplace grid generation equations. The finite element solution procedure and the construction of body conforming coordinates (ξ, η) , have been discussed. The method has been applied for generating grids in simply and multiply connected two dimensional regions. Detailed numerical experimentation has been performed on the construction of O-type grids. and the effects of various features such as the density of boundary grid point data, the shape of the branch cut and the starting guess grid, have been highlighted.

2.2. Literature survey:

Numerical solutions of most engineering problems involve the use of Finite Difference (FD) or Finite Element (FE) methods. For both the methods, the first step concerns with the discretization of the spatial domain into a suitable numerical grid. There are, however, differences in the required grid structure for the two methods.

For FD applications, it is most convenient to consider a regularly placed, rectangular array of grid points. Therefore, a problem with complex geometry is tackled by mapping the

physical domain into a rectangular computational domain, by suitable transformation. Body fitting co-ordinates are employed for transforming the geometry and the implementation of boundary conditions becomes simple when expressed in terms of the body-fitting co-ordinates. Additional requirements on the grid structure for these applications are that the spacing between the grid lines must be controllable (based on the gradients of the solution variables) and the grid lines should not become too skewed.

For FE methods, the role of automatic grid generation is to simplify the specification of input data to the numerical simulation software (pre-processing). It is necessary to number the nodes and elements of the mesh and also generate the data of nodal co-ordinates and the nodal connectivity of elements. Here, the complex geometry of a given solution domain is divided algorithmically into elements of required shape and the other grid data are automatically computed by some algorithms. The grid generation techniques available at present fall into two categories, namely: a) structured grid generation b) unstructured grid generation. Structured grid generation produces a regular array of points and it is suitable for the transformation of a complex geometry into a rectangular one by the use of body-fitting co-ordinates. Therefore, structured grid generation is widely used along with FD simulation software. Unstructured grid generation on the other hand, has no constraints such as the production of a regular array of points. The basic aim here is to approximate a complex geometry in terms of elements of chosen shape, in the best manner possible. The unstructured grid generation, is suited for

FE applications. The earlier work on these generation schemes have been briefly discussed below.

2.2.1. Unstructured grid generation :

Most of the work in unstructured mesh generation relates to techniques such as 2-D quadree and 3-D Octree approaches, and triangulation or sub structuring of the geometry (Batchman et al., 1987; Kela et al., 1986; Shepherd and Yerry, 1983; Joe and Simpson, 1986; Cook, 1981). One of the widely adopted techniques for unstructured grid generation is the automatic triangulation, which involves algorithmic division of the physical region into nearly equilateral triangular elements. Some of these techniques have been extended to produce all-quadrilateral meshes (Shepherd, 1984; Thacker, 1980). Several authors employ parameter space mapping to generate quadrilateral meshes interactively (Cook, 1981; Gordan; 1973; Ho-Le, 1988). Techniques for the local adaptive refinement of the element sizes in required sub regions have also been developed (Yerry and Shepherd, 1988; Zhu and Zienkiewicz, 1991).

2.2.2. Structured grid generation :

The structured grid generation techniques are based on the transformation of the complex physical domain into a simple computational domain, which is often chosen to be rectangular in shape. Body-fitted curvilinear co-ordinates (say ξ and η , where both ξ and η are functions of the physical co-ordinates x and y) are used for the transformation. The numerical grid is developed by plotting the curvilinear co-ordinate lines i.e., $\xi = \text{constant}$

and $\eta = \text{constant}$ curves. The governing equations of the problem to be solved are converted in terms of the curvilinear co-ordinates and the resulting system of equations are solved in the transformed (rectangular) domain by finite difference methods (FDM). The transformation into the ξ - η domain results in an orthogonal and uniform computational grid which is required by FDM. Its advantage lies in the relative easiness of programming since all the calculations are on the regular grid. Moreover, curved boundaries are accounted for and there is no loss of accuracy due to inaccurate modeling of boundary. Further, the grid point positions can be easily changed in the physical domain according to the needs of the computed solution.

Two important questions arise during the generation of body conforming grids. First, how to automatically generate grids which are suited for the problem at hand, and, second, how to control the resulting grid point distribution. In order to address these questions a variety of techniques have been used from simple co-ordinate stretching to the solution of elliptic equations for the curvilinear co-ordinates (Eiseman and Erlebacher; 1987 Thompson, 1982b, 1984, 1985a-d ; Steger and Sorenson, 1979; Smith and Eriksson, 1986). Among these, three important techniques which deserve attention are the algebraic grid generation, conformal mapping and the elliptic PDE solvers. For open (external) domains, parabolic and hyperbolic grid generation methods have been developed using marching techniques (Nakumara, 1982; Dwyer et al., 1982).

A majority of algebraic grid generation techniques employ fast interpolation methods. Among the various interpolation

procedures, transfinite interpolation is a widely used method. In transfinite interpolation, the transformation procedure involves the use of a curvilinear coordinate system which conforms to the shape of the physical region at the boundaries. The interior coordinate lines are obtained using interpolation between the boundary values of the co-ordinates. Grid spacing can also be controlled by prescribing the step size variation on fixed boundaries according to some algebraic expressions (Eiseman, 1986; Eiseman and Erlebacher; 1987; Smith and Eriksson; 1986). These methods are very simple to use. However, for complex geometries, it is difficult to represent the boundary surfaces accurately and the interpolation techniques fail if some parts of the boundary are described by concave surfaces. Handling of sharp corners on the boundary is also not satisfactory.

Complex variable methods which involve conformal mapping have also been widely used (Eiseman and Erlebacher, 1987). Unfortunately they are restricted to two dimensions only and also arbitrary shaped geometries cannot be handled. These techniques have been elaborately discussed in the survey papers, proceedings and the books published by various authors (Eiseman, 1986; Thompson, 1982a,b, 1984; Thompson et al., 1982, 1985a-c; Turkel, 1983; Hawken, 1987).

Thompson et al., (1977, 1982, 1985a-c) have discussed the use of partial differential equations to generate grids in arbitrary geometries. The PDE grid solvers can meet most of the important requirements such as smoothness and orthogonality of grid lines, and control of grid-spacing. Laplace grid generation equations have been considered for obtaining the body-fitting

coordinates during the generation of smooth grids (Thompson et. al., 1985a). These equations have been solved by the finite difference methods over the transformed domain (rectangular) with physical variables x and y as the unknowns. The above authors have derived the orthogonal families of body-fitting curves by solving the Laplace equations. The orthogonal body-fitting grid gives a better representation of a general complex geometry because of its ability to fit a grid line to each boundary surface. However, the transformed equations become quite complex and there is only limited control over the refinement of the generated grid because of the need to retain orthogonality. If orthogonality is sacrificed, then body fitting grids may be derived from the solutions of the Poisson systems. The right hand side of the Poisson equations exercise control over the clustering of grid lines or grid points which is very important from the applications point of view. The use of such control functions on the right hand side allows a very fine resolution at certain specified locations and it covers adaptive grid generation as well. On the other hand, there is a chance that the extremum principles of the Laplace operator may be lost due to the choice of control functions. Many choices for the construction of control functions are discussed in the works of Thompson et al., (1977, 1982, 1985a-c) and Eiseman (1986).

During the transformation of the physical partial differential equations, the gradient, divergence, curl, and Laplace operators get replaced by either their conservative or non-conservative expressions in terms of general curvilinear co-ordinates. Thompson et al., (1985a) have presented the

transformed derivatives (metric coefficients) using the notion of co-variant and contravariant directional derivatives and vector components. From the truncation error point of view, the metric coefficients have to be evaluated accurately. This is because they appear as coefficients for the transformed governing equations of the physical problem as well as the Poisson grid generation system. Thus, grid generation using elliptic PDE is a complex, but very general procedure for obtaining body-fitting grids, with appropriate control for achieving desired grid characteristics. Since the present study deals with the generation of body-fitting grids using an elliptic PDE system, we briefly describe this procedure in the following section.

2.2.3. Grid generation equations based on elliptic equations :

In this subsection, the general two dimensional coordinate transformation procedure and the transformed grid generation equations are presented (Thompson et al. 1985a). These have been widely used in the past for finite difference simulation of physical problems with complex geometries.

Consider a simply or multiply connected region $\Omega(x,y) \subseteq \mathbb{R}^2$ of arbitrary shape to be transformed into a rectangular region $\bar{\Omega}(\xi,\eta)$ as shown in the Figs. 2.1a-b. The general transformation from the physical plane (x,y) to the transformed plane (ξ,η) is given by

$$\xi = \xi(x,y), \quad \eta = \eta(x,y). \quad (2.1a)$$

Similarly, the inverse transformation is given by

$$x = x(\xi,\eta), \quad y = y(\xi,\eta). \quad (2.1b)$$

For a given function $f \equiv f(x,y)$, the transformed derivatives can

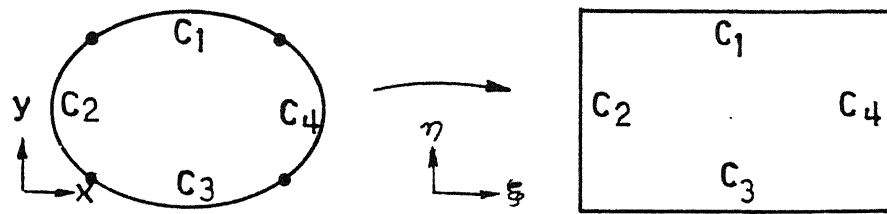


Fig. 2.1(a) Transformation of an ellipse to a rectangle.

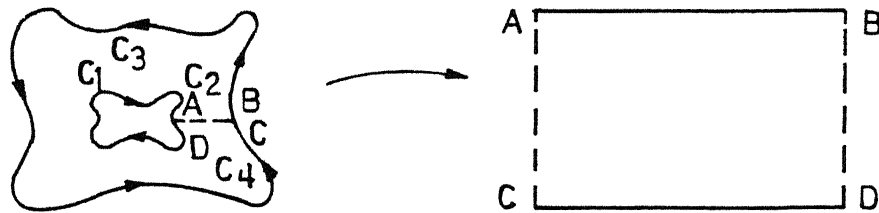


Fig. 2.1(b) Transformation of an annular region to a rectangle.

be written as follows :

$$\begin{aligned} f_x &= \frac{\partial(f, y)}{\partial(\xi, \eta)} / \frac{\partial(x, y)}{\partial(\xi, \eta)} = \left(\frac{\partial y}{\partial \eta} \frac{\partial f}{\partial \xi} - \frac{\partial y}{\partial \xi} \frac{\partial f}{\partial \eta} \right) / J \\ f_y &= \frac{\partial(f, x)}{\partial(\xi, \eta)} / \frac{\partial(x, y)}{\partial(\xi, \eta)} = \left(-\frac{\partial x}{\partial \eta} \frac{\partial f}{\partial \xi} + \frac{\partial x}{\partial \xi} \frac{\partial f}{\partial \eta} \right) / J \end{aligned} \quad (2.2)$$

where J is the Jacobian of transformation given by ,

$$J = \frac{\partial x}{\partial \xi} \frac{\partial y}{\partial \eta} - \frac{\partial x}{\partial \eta} \frac{\partial y}{\partial \xi} \quad (2.3)$$

The basic idea of the transformation is to generate transformation functions such that all boundaries are coincident with co-ordinate lines. The body-fitting co-ordinates (ξ, η) are taken as the solutions of a suitably chosen elliptic boundary value problem, with one of the co-ordinates constant on the boundaries. The Poisson equations associated with grid generation can be written as

$$\frac{\partial^2 \xi}{\partial^2 x} + \frac{\partial^2 \xi}{\partial^2 y} = P(\xi, \eta) \quad (2.4a)$$

$$\frac{\partial^2 \eta}{\partial^2 x} + \frac{\partial^2 \eta}{\partial^2 y} = Q(\xi, \eta) \quad (2.4b)$$

where $x = x(\xi, \eta)$, $y = y(\xi, \eta)$ and $x, y \in \Omega \subseteq \mathbb{R}^2$ with suitable boundary conditions which specify either the grid spacing or the grid slope at the boundary. Interchanging dependent and independent variables for eqns. (2.4a-b) gives :

$$\alpha \frac{\partial^2 x}{\partial \xi^2} - 2\beta \frac{\partial^2 x}{\partial \xi \partial \eta} + \gamma \frac{\partial^2 x}{\partial \eta^2} + J^2 \left(P \frac{\partial x}{\partial \xi} + Q \frac{\partial x}{\partial \eta} \right) = 0 \quad (2.5a)$$

$$\alpha \frac{\partial^2 y}{\partial \xi^2} - 2\beta \frac{\partial^2 y}{\partial \xi \partial \eta} + \gamma \frac{\partial^2 y}{\partial \eta^2} + J^2 \left(P \frac{\partial y}{\partial \xi} + Q \frac{\partial y}{\partial \eta} \right) = 0 \quad (2.5b)$$

$$\text{where } \alpha = \left(\frac{\partial x}{\partial \eta} \right)^2 + \left(\frac{\partial y}{\partial \eta} \right)^2; \quad \beta = \frac{\partial x}{\partial \xi} \frac{\partial x}{\partial \eta} + \frac{\partial y}{\partial \xi} \frac{\partial y}{\partial \eta} \quad \text{and} \quad \gamma = \left(\frac{\partial x}{\partial \xi} \right)^2 + \left(\frac{\partial y}{\partial \xi} \right)^2$$

The equations (2.5a-b) can now be solved on the transformed ξ - η plane using finite difference methods, and the values of the physical co-ordinates (x, y) for each grid point (ξ, η) in the transformed plane can be obtained. It is seen that the equations in the transformed plane are more complicated than in the physical plane, however with the advantage that the solutions are being found for a simple square grid. Use of the boundary-fitted co-ordinate system enables the boundary conditions to be applied easily.

2.3. Scope of present work :

In the previous section, the generalized approach of elliptic PDE based grid generation procedure was outlined. It can be shown (Thompson et. al. 1985; Saltzman and Brackbill, 1982) that the Poisson grid generation equations correspond to the minimization of a functional which is made up of terms related to grid smoothness, orthogonality and spacing control. In this sense, the elliptic PDE solvers can provide optimal grids, for a given physical problem. However, as discussed in the above section, when the grid generation equations are transformed in terms of the body fitting coordinates for facilitating FD solutions, the resulting equations are highly nonlinear and complex. These have to be solved iteratively and the iterative procedure often poses serious difficulties for convergence.

In the present work, structured grid generation by the finite

element solution of elliptic partial differential equations has been studied*. The generation of structured grids in the physical domain by finite element method can offer a lot of advantages. This is due to fact that the original Poisson grid generation equations (2.4 a, b) are linear and these can be solved easily in the physical domain itself. Consequently important grid features such as orthogonality, smoothness etc. can be preserved, since no non-linear iterations are involved. Most numerical algorithms are very sensitive to grid skewness, clustering and smoothness; if grid generation procedure does not converge properly, the evaluation of metric terms may affect the numerical accuracy of the predicted solutions for such schemes. Thus, there is a strong need to simplify the solution procedure for the elliptic PDE solvers and develop rapidly convergent grid generation schemes.

The generation of smooth body fitting grids in complex geometries using Laplace equations is considered in the present work. Since all the computations are performed in the physical domain only, there is no need to transform the grid generation equations in terms of the curvilinear co-ordinates. Apart from linearity, the other inherent advantages of the Laplacian grid solver are also intactly preserved. The implementation of Dirichlet and Neumann boundary conditions which relate to the grid spacing and the slope of grid lines at the boundary respectively, is easily achieved in the proposed scheme.

2.4. Geometry of the problems considered :

Description of the Geometry :

The present method has been applied to a simply connected

* This work is published in the conference (see, Rao et al. 1988).

domain (Fig. 2.1a) and a multiply connected region with an O-type grid (Fig. 2.1b). In the former case, the transformation is from an ellipse to a rectangular region and in the latter, it is from an annular region to a rectangle with the use of a branch cut. Different grid configurations of the body-fitting co-ordinate system can be created by identifying different boundary segments of the physical region with particular sides of the transformed rectangular region (Thompson et al., 1985a).

Boundary conditions :

The boundary conditions for the elliptic domain problem are of Dirichlet type (Fig. 2.1a). In this case the boundary of the ellipse is divided into two ξ - constant curves (C_2 and C_4) on which η varies monotonically and two η - constant curves (C_1 and C_3) on which ξ varies monotonically. For the annular region (Fig. 2.1b), the ξ - variation is specified on the η constant curves C_1 and C_3 . On the ξ - constant curves C_2 and C_4 , the η -variation is left unspecified and re-entrant boundary conditions for η are enforced. In the finite element implementation, the re-entrant boundary conditions are applied very easily by declaring the respective elements and nodes across the branch cut to be neighbours (by specifying the nodal connectivity of elements appropriately). Apart from Dirichlet boundary conditions for ξ on C_1 and C_3 , Neumann boundary conditions (the gradient of ξ in the direction normal to the boundary, $\frac{\partial \xi}{\partial n}$) can also be specified depending upon the need. A particular case of interest is the orthogonal grid at the boundary; for this, the value of the gradient, $\frac{\partial \xi}{\partial n}$ is taken as zero.

2.5. Finite element formulation of grid generation equations :

The grid generation equations which have been used in the present work are of the form,

$$\frac{\partial^2 \xi}{\partial x^2} + \frac{\partial^2 \xi}{\partial y^2} = 0 \quad (2.6a)$$

$$\frac{\partial^2 \eta}{\partial x^2} + \frac{\partial^2 \eta}{\partial y^2} = 0 \quad (2.6b)$$

where $x = x(\xi, \eta)$, $y = y(\xi, \eta)$ and $x, y \in \Omega \subseteq \mathbb{R}^2$, with the boundary conditions specified as mentioned above.

Finite element discretization :

The given physical domain $\Omega(x, y)$ is divided into several small quadrilateral elements (Figs 2.2 and 2.3). Eight-noded curvilinear elements have been employed in the present work, which can fit arbitrary geometries quite well. Isoparametric mapping of each curvilinear element has been performed with the help of local curvilinear coordinates \bar{x} and \bar{y} , as shown in Fig. 2.2. As per conventional FE procedure, such local mapping facilitates numerical quadrature during the calculation of elemental matrices, corresponding to the discretized forms of the grid generation equations (2.6a,b).

The construction of the numerical grid using ξ - and η -constant curves can be explained in the following manner. We assume that on any ξ curve for which η is constant, there are $N(\xi)$ grid points and on any η curve (for which ξ is constant) there are $N(\eta)$ grid points. Consider any curve $\xi = \xi^*$, which lies between the boundary curves $\xi = \xi_{min}$ and $\xi = \xi_{max}$ of the

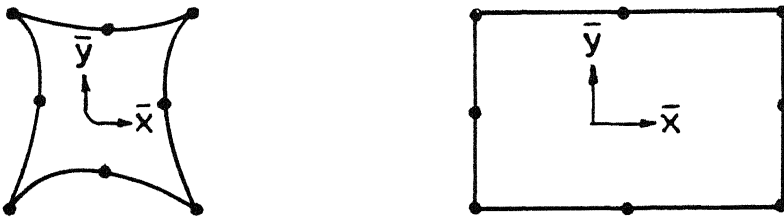


Fig.2.2 Eight noded isoparametric element and local coordinate system.

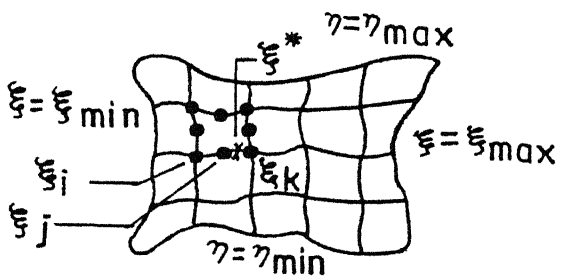


Fig.2.3 Construction of ξ -constant lines.

domain. Now, on the curve $\xi = \xi^*$, the grid points are connected by a piece-wise continuous curve which is made up of the boundary segments of the quadrilateral elements, as shown in Fig. 2.3. When Laplace equations are solved, the grid formed by $\xi = \text{constant}$ and $\eta = \text{constant}$ curves is very smooth i.e., even the slopes of the piece-wise segments match at the grid points along the coordinate lines. Now we proceed to present the exact procedure followed in actual computation.

The integral statements of the governing PDE'S for grid generation can be set up by multiplying eq. (2.6a,b) by test functions v^ξ and v^η respectively and integrating over the domain Ω . After integration by parts, the resulting equations for weak formulation are:

$$\iint_{\Omega} \left\{ \frac{\partial v^\xi}{\partial x} \frac{\partial \xi}{\partial x} + \frac{\partial v^\xi}{\partial y} \frac{\partial \xi}{\partial y} \right\} dx dy - \int_{\partial\Omega} v^\xi \frac{\partial \xi}{\partial n} dl = 0 \quad (2.7a)$$

$$\iint_{\Omega} \left\{ \frac{\partial v^\eta}{\partial x} \frac{\partial \eta}{\partial x} + \frac{\partial v^\eta}{\partial y} \frac{\partial \eta}{\partial y} \right\} dx dy - \int_{\partial\Omega} v^\eta \frac{\partial \eta}{\partial n} dl = 0 \quad (2.7b)$$

where n is the coordinate along the outward normal to the boundary $\partial\Omega$. According to the Galerkin's approach, the test functions v^ξ , v^η and the solution variables ξ and η , can be taken to belong to the same finite dimensional solution space. For instance, both may be represented by piece-wise bilinear polynomial functions. Representing the finite dimensional approximations by an over bar, eq. (2.7a,b) can be rewritten as:

$$\iint_{\Omega_h} \left\{ \frac{\partial \bar{V}^\xi}{\partial x} \frac{\partial \bar{\xi}}{\partial x} + \frac{\partial \bar{V}^\xi}{\partial y} \frac{\partial \bar{\xi}}{\partial y} \right\} dx dy - \int_{\partial \Omega_h} \bar{V}^\xi \frac{\partial \bar{\xi}}{\partial n} dl = 0 \quad (2.8a)$$

$$\iint_{\Omega_h} \left\{ \frac{\partial \bar{V}^\eta}{\partial x} \frac{\partial \bar{\eta}}{\partial x} + \frac{\partial \bar{V}^\eta}{\partial y} \frac{\partial \bar{\eta}}{\partial y} \right\} dx dy - \int_{\partial \Omega_h} \bar{V}^\eta \frac{\partial \bar{\eta}}{\partial n} dl = 0 \quad (2.8b)$$

Here, the functions \bar{V}^ξ , \bar{V}^η , $\bar{\xi}$ and $\bar{\eta}$ lie in the appropriate finite dimensional solution space (say S_h), while Ω_h is an approximation to Ω , with $\partial \Omega_h$ as its boundary. For convenience sake, we denote the finite dimensional approximation $\bar{\xi}$ as ξ and $\bar{\eta}$ as η respectively. Let N_h be the dimension of the finite dimensional solution space, S_h . Expressing ξ , η in terms of basis functions ψ_i , $1 \leq i \leq N_h$; we have

$$\xi = \sum_{i=1}^{N_h} \psi_i(x,y) \xi_i, \quad \eta = \sum_{i=1}^{N_h} \psi_i(x,y) \eta_i \quad (2.9)$$

As per the standard Galerkin procedure, we assume the test functions to be same as the basis functions. Thus,

$$V_i^\xi = \psi_i(x,y)$$

$$\text{and } V_i^\eta = \psi_i(x,y), \quad 1 \leq i \leq N_h$$

Now, equations (2.8a-b) lead to the following matrix equations

$$[C] \{ \xi \} = \{ d^1 \} \quad (2.10a)$$

$$[C] \{ \eta \} = \{ d^2 \} \quad (2.10b)$$

where, $C = [C_{ij}]$

is the matrix with entries $C_{ij} = \iint_{\Omega_h} \left(\frac{\partial \psi_i}{\partial x} \frac{\partial \psi_j}{\partial x} + \frac{\partial \psi_i}{\partial y} \frac{\partial \psi_j}{\partial y} \right) dx dy$

$$1 \leq i \leq N_h, \quad 1 \leq j \leq N_h$$

and $\{d^1\}$ and $\{d^2\}$ are the right hand side vectors whose non-zero

entries arise from the Dirichlet or Neumann boundary conditions. From eq. (2.8a,b), it is evident that taking the right hand side contribution to be zero (for ξ variable on the $\eta = \text{constant}$ boundary and vice versa), enforces orthogonality condition at the boundary. Since the elements are isoparametric and eight-noded, both the physical coordinates (x,y) and the transformed coordinates (ξ, η) can be interpolated within each element via the expressions

$$x = \sum_{i=1}^8 \psi_i(\bar{x}, \bar{y}) x_i \quad y = \sum_{i=1}^8 \psi_i(\bar{x}, \bar{y}) y_i \quad (2.11a)$$

$$\xi = \sum_{i=1}^8 \psi_i(\bar{x}, \bar{y}) \xi_i \quad \eta = \sum_{i=1}^8 \psi_i(\bar{x}, \bar{y}) \eta_i \quad (2.11b)$$

where (\bar{x}, \bar{y}) denote the local coordinates. The relationships between lengths can be obtained as follows:

Setting $x \equiv x(\bar{x}, \bar{y})$, $y \equiv y(\bar{x}, \bar{y})$

$$dx = \frac{\partial x}{\partial \bar{x}} d\bar{x} + \frac{\partial x}{\partial \bar{y}} d\bar{y} ; dy = \frac{\partial y}{\partial \bar{x}} d\bar{x} + \frac{\partial y}{\partial \bar{y}} d\bar{y} \quad (2.12)$$

or

$$\begin{bmatrix} dx \\ dy \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial \bar{x}} & \frac{\partial x}{\partial \bar{y}} \\ \frac{\partial y}{\partial \bar{x}} & \frac{\partial y}{\partial \bar{y}} \end{bmatrix} \begin{bmatrix} d\bar{x} \\ d\bar{y} \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^8 \frac{\partial \psi_i}{\partial \bar{x}} x_i & \sum_{i=1}^8 \frac{\partial \psi_i}{\partial \bar{y}} x_i \\ \sum_{i=1}^8 \frac{\partial \psi_i}{\partial \bar{x}} y_i & \sum_{i=1}^8 \frac{\partial \psi_i}{\partial \bar{y}} y_i \end{bmatrix} \begin{bmatrix} d\bar{x} \\ d\bar{y} \end{bmatrix} \quad (2.13)$$

In a similar fashion , the derivatives can be obtained as

$$\begin{bmatrix} \frac{\partial}{\partial \bar{x}} \\ \frac{\partial}{\partial \bar{y}} \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial \bar{x}} & \frac{\partial y}{\partial \bar{x}} \\ \frac{\partial x}{\partial \bar{y}} & \frac{\partial y}{\partial \bar{y}} \end{bmatrix} \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{bmatrix} = [J] \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{bmatrix} \quad (2.14)$$

where $[J]$ is the Jacobian matrix for the transformation which can be evaluated using equations (2.13). Making use of (2.14) and Gauss-Legendre quadrature formulae, the left hand and right hand sides of the matrix equations (2.10) are evaluated for each element. The expressions for the basis functions ψ_i are available in most of the books on the finite element method (Ciarlet, 1978; Oden and Reddy, 1976).

Solution procedure : During the actual computations, the frontal method is used to solve the matrix system arising from the FE approximation of the grid generation equations. It is well known that the method adopted for solving the assembled matrix equation has a significant bearing on the computer storage requirement and execution time when the total number of unknown variables is very large. The frontal solution technique is based on the direct Gaussian elimination procedure for solving symmetric matrices where the leading diagonal is used as pivot. The details of the Frontal technique have been discussed (Irons, 1970; Taylor and Hughes, 1981).

2.6. Description of the algorithm :

In this section, the algorithm for the numerical generation of a structured, smooth body-fitting grid inside an arbitrary shaped region has been described. For computing the FE solution, a guess grid is provided initially using eight-noded, isoparametric, quadrilateral elements. The transfinite interpolation technique is used for the generation of the initial grid. With the initial mesh, the grid generation equations are

solved and the values of ξ and η are predicted at all the grid points of the guess grid. Now, the construction of ξ and η constant lines is performed by the use of interpolation in the physical domain. The new grid locations x and y are calculated for each grid point by determining the intersection points of ξ and η constant grid lines. Since the initial guess grid may not possess the desired grid properties, the predicted values of ξ and η (on the guess grid) may not be very accurate. Therefore, the entire process of solving the ξ -problem and the η -problem is repeated for one or two iterations with the new grid locations (x,y) obtained each time. With each successive iteration, the grid manifests more conformity with the given geometry (except at sharp corners) and the grid lines become smoother. After a few of iterations, the grid lines closely approximate ξ -constant and η -constant lines and at this stage, the finally converged grid is obtained. These steps of the algorithm are summarized below. The corresponding flow chart is given (Fig. 2.4).

step. 1 : Define initial data on the given geometry

Define the boundary of the physical domain $\Omega(x,y)$,
 Define ξ and η variation on the boundaries of $\Omega(x,y)$ and
 Normally, on each η - constant boundary, equal increment
 in ξ is considered between any two neighbouring grid
 points, and vice versa.

step. 2 : Calculate initial grid on $\Omega(x,y)$

Obtain the initial grid points $x_{old}(i)$ $y_{old}(i)$
 $i = 1, \dots, N_h$ with the specified boundary data on
 $\partial\Omega$, by transfinite interpolation.

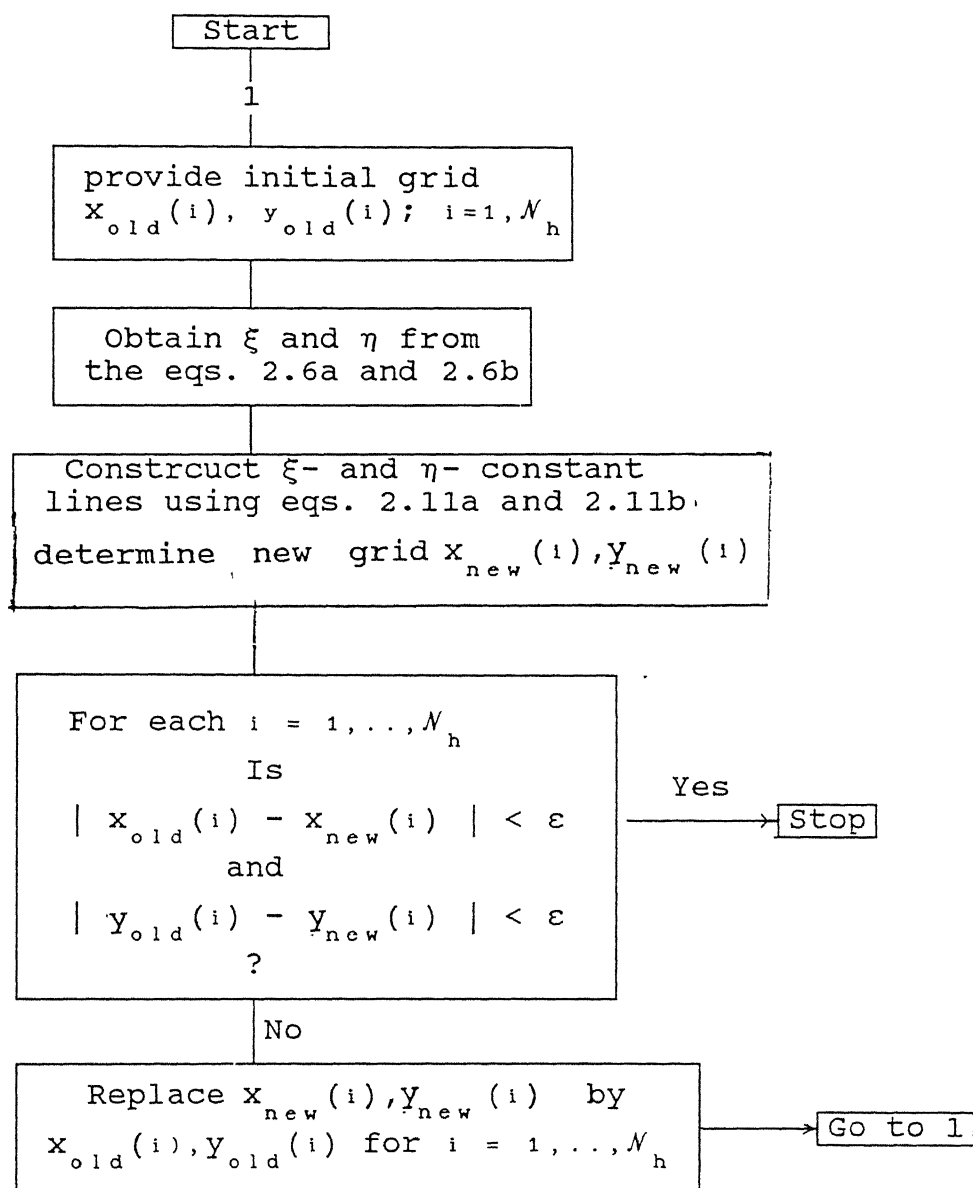


Fig. 2.4. Flow chart for the generation of the grid.

Step. 3 : Solve for body-fitting co-ordinates (ξ, η)

Solve the grid generation equations (2.6a-b)

by finite element method with the boundary conditions specified as above in the physical domain.

Step. 4. Construct ξ -constant and η -constant lines

Construct $\xi = \xi_i^*$ ($2 \leq i \leq N(\xi)-1$) and $\eta = \eta_j^*$ ($2 \leq j \leq N(\eta)-1$)

curves by finite element interpolation and

determine new grid (x_{new}, y_{new}) locations of each ξ_i^*, η_j^*). Here, ξ_i^* and η_j^* are set equal to the values of ξ and η on the boundary.

Step. 5. convergence criterion for iterations

If $\left(\max_{1 \leq i \leq N_h} | (x_{old}(i) - x_{new}(i)) | \right) \leq \epsilon$ (a prescribed tolerance)

and

$\left(\max_{1 \leq i \leq N_h} | (y_{old}(i) - y_{new}(i)) | \right) \leq \epsilon$

then stop the iterative process;

otherwise go to Step 6.

Step. 6. Perform grid iteration

Replace the old grid values (x_{old}, y_{old}) by new grid values (x_{new}, y_{new}) and go to step 3.

2.7. Important features of the algorithm :

In this section , the important features of the algorithm such as generation of the initial grid, construction of ξ and η constant lines etc. have been discussed.

Algebraic grid generation system

Generation of the curvilinear coordinate system (ξ, η) can be formulated as a problem of finding the cartesian coordinates

(x,y) in the interior of the rectangular transformed region from their specified values on the boundaries. This can be done by interpolation between the specified boundary values of (x,y) , for any interior point depending on its location (ξ,η) in the transformed domain. The basic problem, therefore, consists of finding suitable interpolation functions for generating the interior co-ordinates of the physical domain. A large volume of literature is available on such algebraic grid generation methods and these have been discussed in the surveys by Thompson et al., 1982, 1985a; Ghia and Ghia, 1983; Eiseman and Eriebacher, 1987.

In the present work, transfinite interpolation (Thompson et al., 1985a) has been used to construct the initial grid. The bi-directional interpolation is implemented in two unidirectional interpolation steps; interpolation is first performed one direction (say η) over the entire domain and then it is applied in the other direction (ξ), taking due care to match the prescribed boundary values of co-ordinates. The procedure is simple to use and is inexpensive computationally. It can also be extended easily to three-dimensional situations .

Construction of ξ and η -constant lines :

The construction of ξ - and η -constant lines is done as follows. For obtaining the shape of the curve $\xi = \xi^*$, the predicted values of the variable ξ over the whole domain are scanned, to identify the elements on whose boundary ξ^* occurs. If for two neighbouring nodes of an element (say j and k) $\xi_j \leq \xi^* \leq \xi_k$, then the $\xi = \xi^*$ curve passes between the nodes j and k (Fig. 2.3). All such elements and the corresponding boundary segments are considered for the construction of the $\xi = \xi^*$ curve. Now on

each of these boundary arcs, it is necessary to find the physical co-ordinates (x,y) at the locations where $\xi = \xi^*$ occurs.

Consider a typical elemental boundary arc, on which ξ^* occurs. Depending on which side of the element it corresponds to, one of the local coordinates (\bar{x} or \bar{y}) will be constant on the arc (see Fig. 2.2). If \bar{x} is the local coordinate which varies along the elemental arc (with \bar{y} being constant on the arc), then the interpolation specified by equation (2.11a) simplifies to the form

$$\xi = a \bar{x}^2 + b \bar{x} + c$$

where

$$a = (\xi_1 + \xi_k - 2\xi_j) / 2; \quad b = (\xi_k - \xi_1) / 2; \quad c = \xi_j$$

and i, j, k are the numbers of the nodes which constitute the boundary arc. Solving the quadratic equation given by

$$a \bar{x}^2 + b \bar{x} + c - \xi^* = 0$$

the local coordinate \bar{x} can be determined. Since \bar{y} has a known constant value on the boundary arc, the physical coordinates (x,y) of the point at which $\xi = \xi^*$, can be found from equations (2.11b).

2.8. Results and Discussion :

The method has been applied for generating grids in simply and multiply connected two dimensional regions. Effects of boundary spacing and initial grid structure have been studied and the findings are discussed below.

In all the Figs. 2.5-2.11, the intersection of ξ - and η -constant lines in the final grid are totally different from the

initial grid. Within the first one or two iterations, smooth grids conforming to the geometry of the problem are obtained, irrespective of the initial grid structure. Due to the extrema principles of the partial differential equation system in the physical plane and the monotone variation of body-fitting co-ordinates along the boundary segments, it is clear that overlapping grid lines will not occur in the physical domain. Further, due to the inherent smoothness of the Laplace operators, whatever grid roughness present in the initial grid, disappears after the application of the algorithm.

The grid generated in a simply connected C shaped domain is shown in the Fig. 2.5. The domain boundaries are smooth with a mild curvature in one direction, while they are non-smooth and possess sharp curvatures in the other direction. Starting with a rough initial grid, within few iterations, excellent smoothness is achieved in the domain interior. Another point to be noted is that in the direction of mild curvature, the grid spacing remains more or less uniform; in the other direction, grid points move away from concave corners towards convex corners. This is a well documented property of the elliptic grid generation systems (Thompson et al., 1985a). Only a few ξ and η grid lines have been considered for the sake of simplicity, during the generation of grids. By incrementing the number of grid lines where required, grid smoothness can be improved and grid spacing can also be controlled.

In the Figs. 2.6 and 2.7, grids for some simply connected domains have been shown. In one case, a uniform grid spacing on the boundary is considered, while in the other, dense

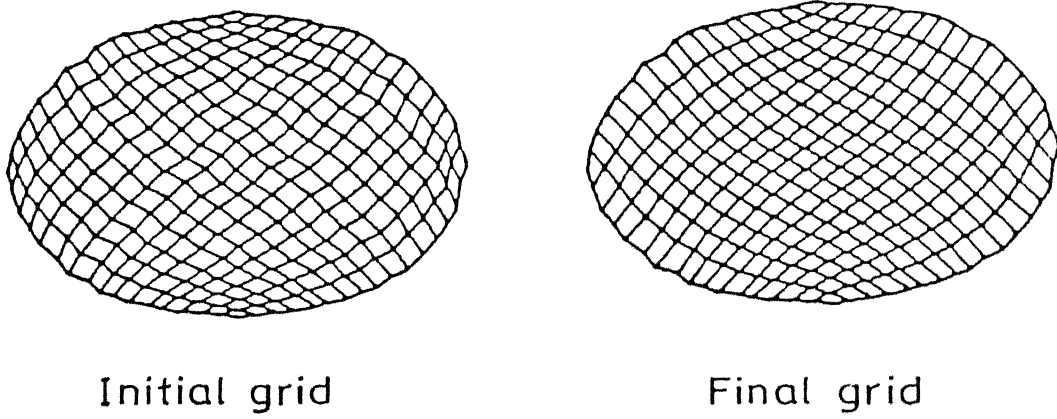
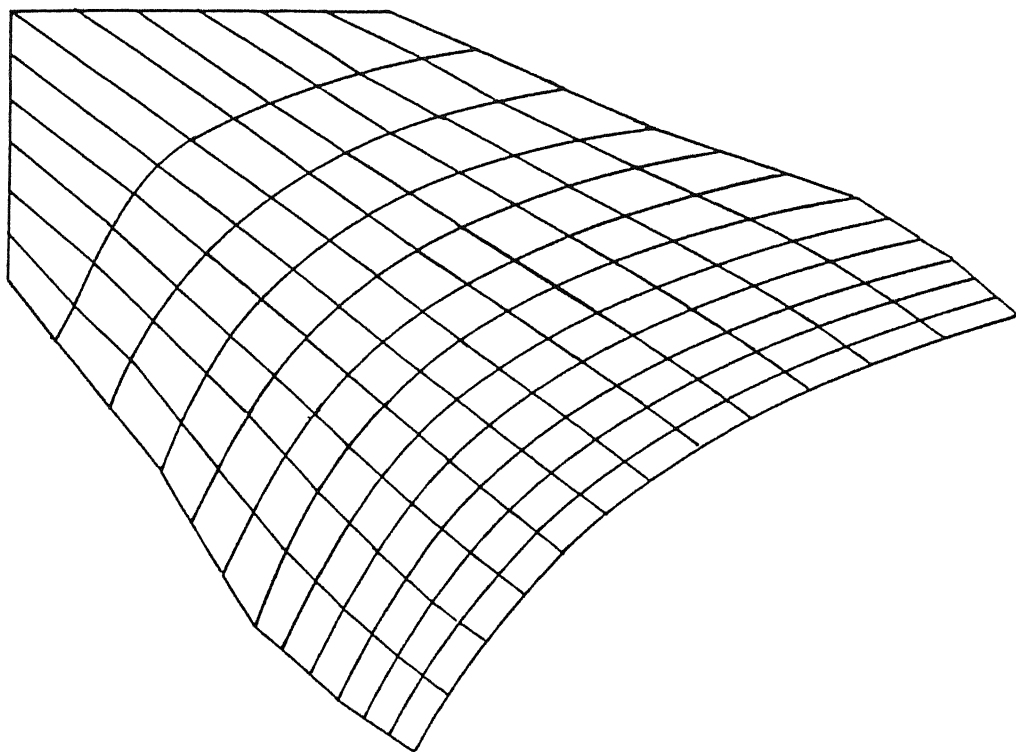
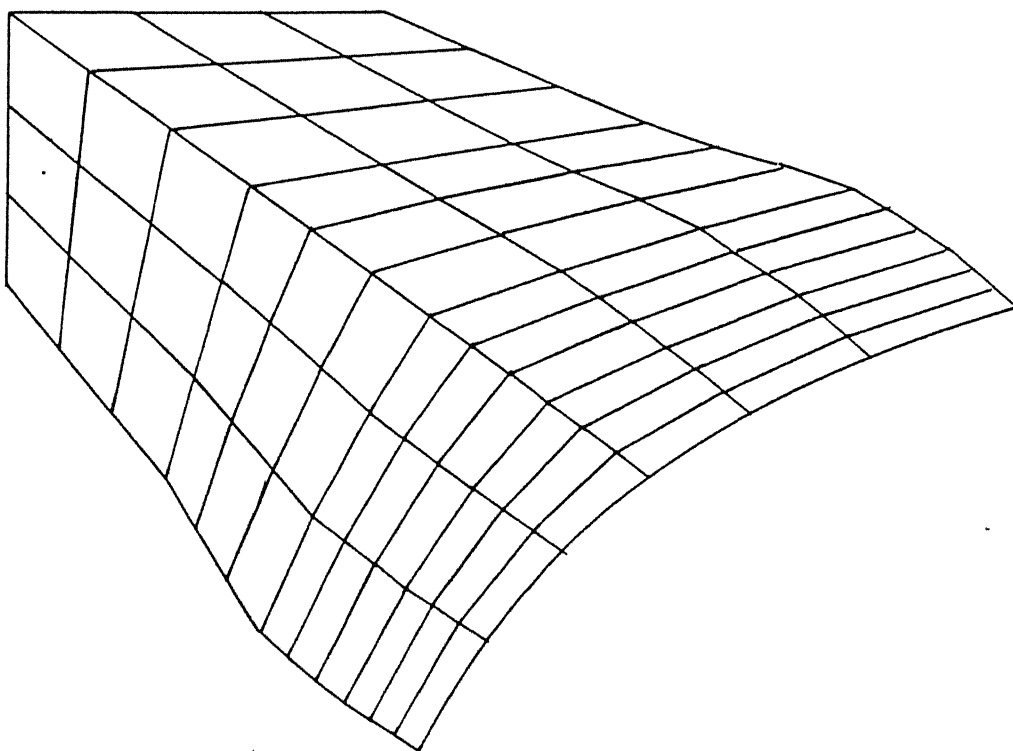


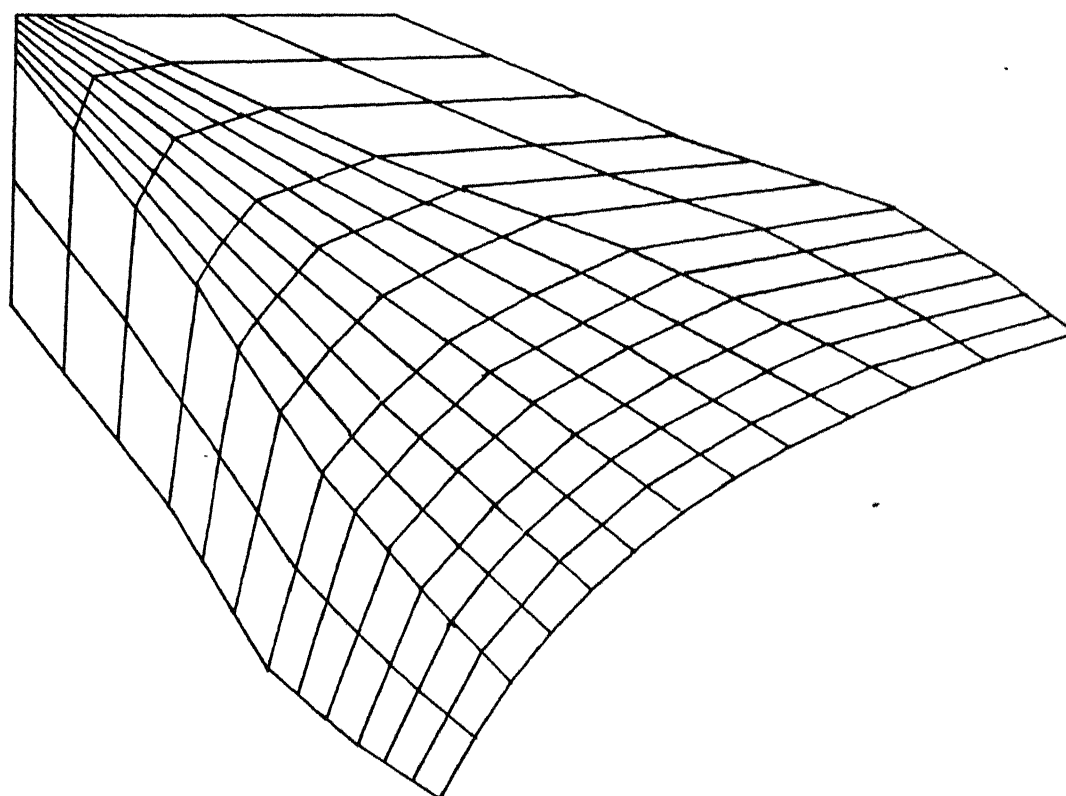
Fig. 2.5 Numerical grid for the ellipse domain.



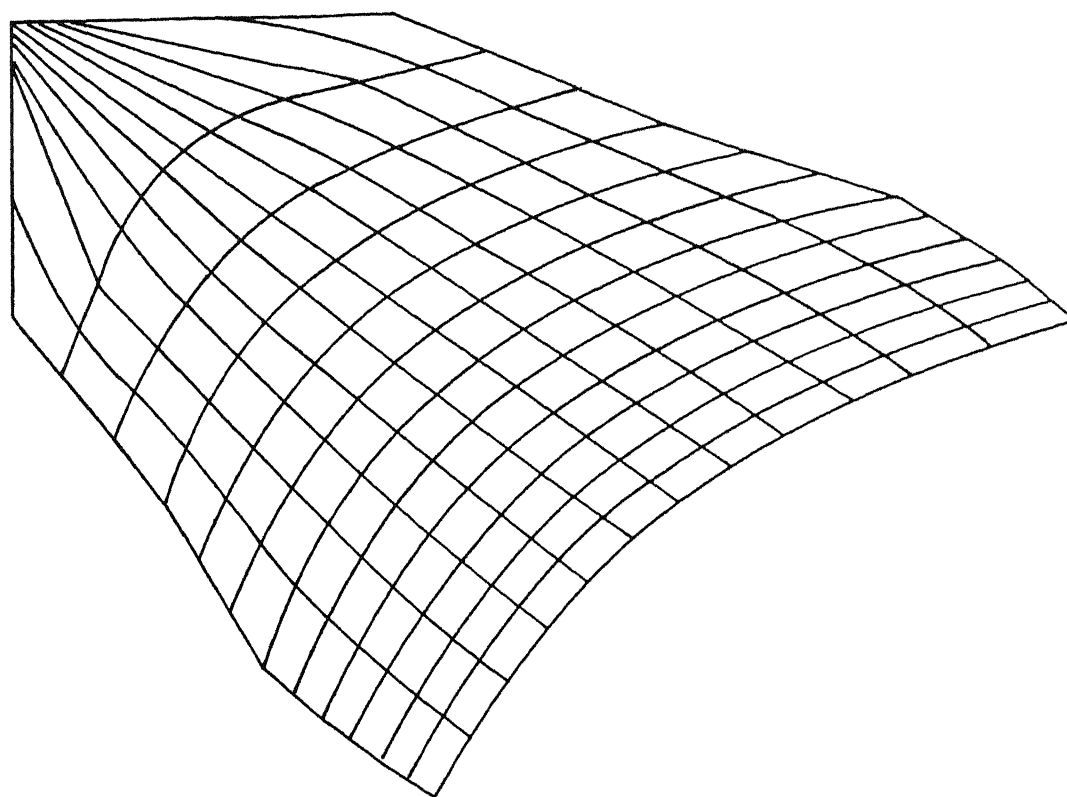
Initial grid

Final grid

Fig. 2.6 Numerical grid for the simply connected domain :
Effect of uniform grid spacing.



Initial grid



Final Grid

Fig. 2.7 Numerical grid for the simply connected domain:
Effect of dense boundary data .

specification of boundary data near the sharp corner of a side has been considered. The smoothness of interior grid lines and the behaviour near concave and convex boundaries after the application of the Laplace solver are evident from the figures. The important observation from these figures is that although the initial grids are totally different from each other, the converged grids have good correspondence, especially in regions where the boundary data remains same.

The effect of boundary roughness on the grid within an elliptical domain is shown in Fig. 2.8. The boundaries of the domain have been made rough purposefully, to study the smoothening effect of the present algorithm. The transfinite interpolation produces a rough initial grid at interior points also. However, after the application of the PDE based grid generation approach for four iterations, the grid lines become very smooth in the interior. As discussed earlier, such grid smoothness is essential for obtaining accurate numerical solutions to physical problems. It is also seen that the grid spacing is uniform in the interior portions, which is a well known property of the Laplace system.

In Fig. 2.9. the effect of dense boundary grid point data, on the O-type grid within an annular region is shown. Due to the convex property of the Laplacian operator, the grid lines tend to move away from concave corners, towards convex points. The closed η coordinate lines show body-conformity except at sharp corners. The ξ coordinate lines, however, are distorted due to the (x-y) symmetry of the problem and the close spacing of the boundary points specified at the outer boundary of the domain.

For uniformly distributed boundary data, both ξ and η lines

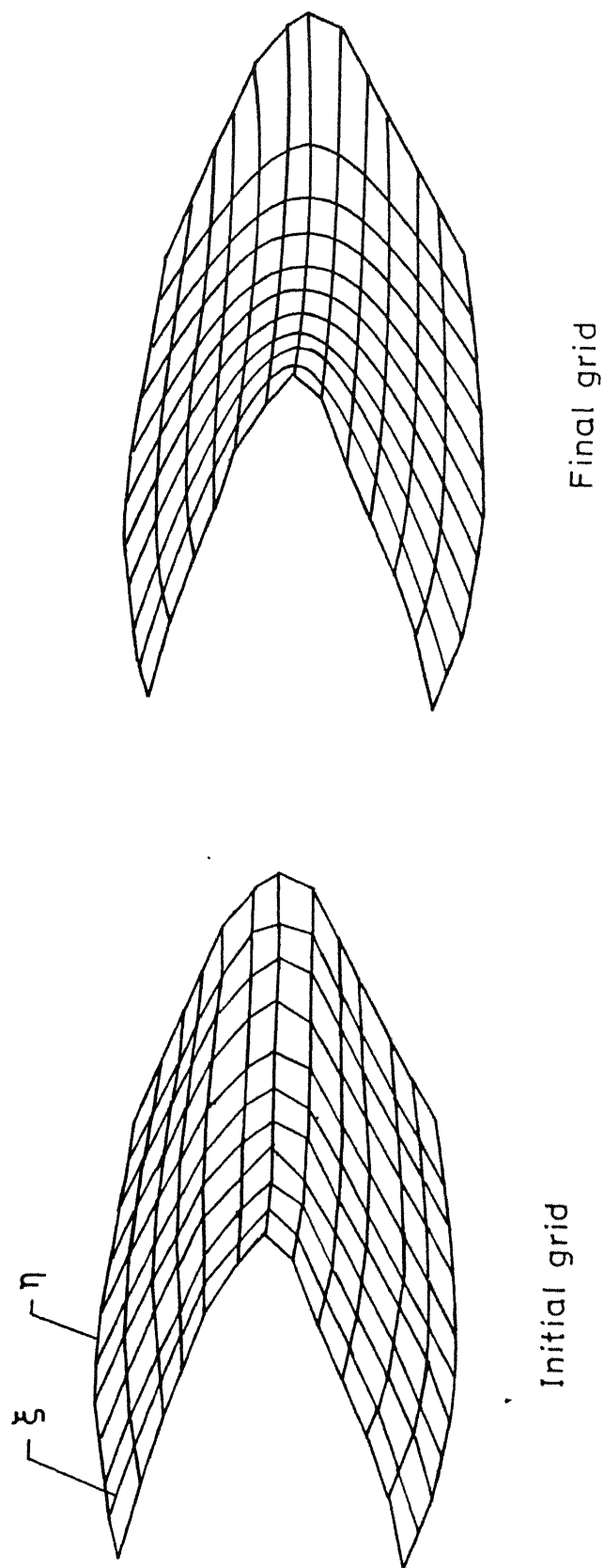


Fig.2.8 Numerical grid for the simply connected domain :
Effect of non - smooth boundaries .

A---B : branch cut

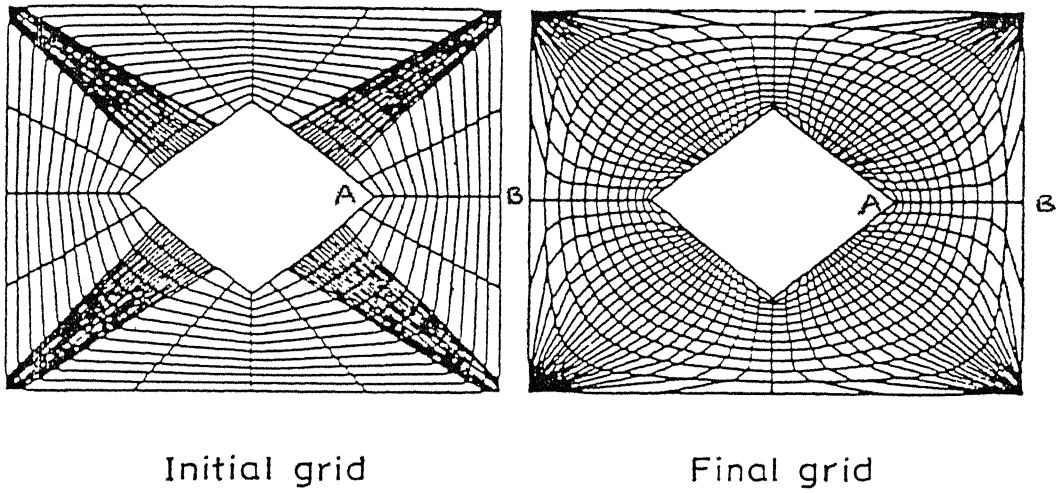
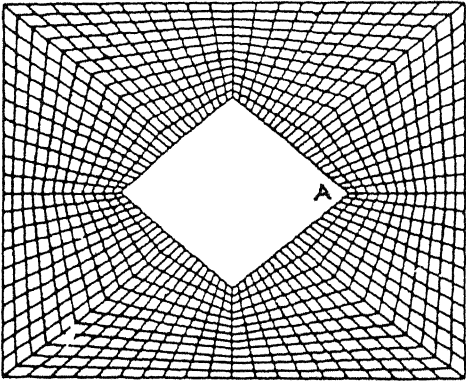
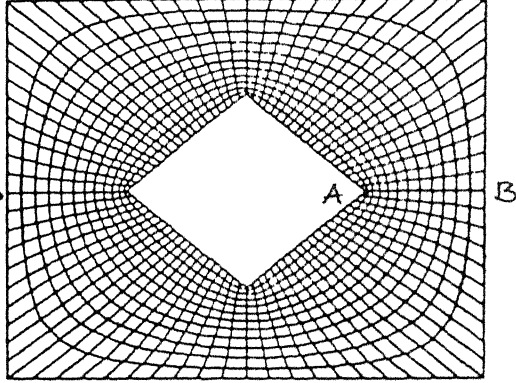


Fig.2.9 O-type grid for dense boundary data around concave corners.

A---B : branch cut



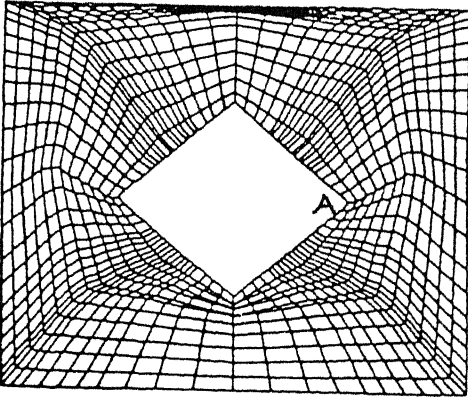
Initial grid



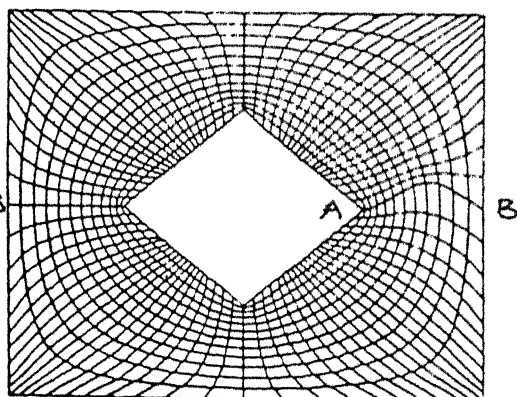
Final grid

Fig. 2.10 O-type grid for uniformly distributed boundary data.

A_γ---B : branch cut



Initial grid



Final grid

Fig. 2.11 O-type grid for arbitrary shaped branch cut with a very bad initial guess.

are smooth and they do not display much distortion, as seen in Fig. 2.10. The effects of a very bad initial grid is shown in Fig. 2.11. The present technique was found to be very stable during the grid smoothening process. However, at times, local distortion of elements occurred due to the rapid movement of points near concave corners. This problem was resolved by under relaxation. It is also seen from Fig. 2.11 that a complicated shape of branch cut influences the grid only locally and the effect is forgotten further away.

2.9. Conclusions :

In this chapter, the Laplace grid generation equations have been used to generate body-fitting grids in arbitrary-shaped bodies. Finite element technique has been employed to solve the grid generation equations in the physical domain itself. The work described here has a greater flexibility to account for various types of boundary conditions. It is also a rugged procedure because of the fact that only linear equations have been used to obtain the required grid. It must be noted that local refinement schemes can be easily implemented by choosing appropriate control functions in Poisson grid generation equations and the present method can be extended to adaptive grid generation. Extension to 3-D grid generation is also possible.

CHAPTER 3

FINITE ELEMENT FORMULATION OF FLAME PROPAGATION PROBLEMS AND
DESCRIPTION OF COMPUTABLE ERROR ESTIMATORS FOR GRID ADAPTION

3.1. Introduction :

With the advent of the energy crisis, combustion studies have assumed great significance, with a view towards making the combustion process fuel- efficient. A majority of the theoretical studies on combustion involve numerical simulation, due to the inherently non-linear nature of the reaction processes. Central to such numerical simulation of combustion phenomena, is the capability to track the movement of a flame front. Since the flame is a zone of intense heat release and rapid chemical reactions, it is characterized by steep gradients of temperature and species concentrations. Obviously, adaptive mesh generation is an essential feature of any numerical simulation technique for the efficient solution of flame propagation problems.

In the ensuing chapters, algorithms for generating adaptive meshes and solving front propagation problems are discussed. In the present chapter, general features of the finite element formulation and the solution procedure for a one dimensional reactive flow system are described. Further, the mathematical preliminaries required for defining various measures of computable error and the use of these error measures in conjunction with the finite element method are presented.

3.2. Modelling of flame propagation problem :

3.2.1. General background :

The main quantities of interest in any combustion study are the global flame properties such as the flame length and the overall heat release, the profiles of temperature and species concentrations and the flow features like turbulent mixing. Detailed theoretical prediction of these quantities is a formidable task, due to a variety of reasons. For instance, flames typically involve a large number of species and free radicals whose rates of production and consumption are governed by several simultaneous reactions occurring at different speeds. Another major complexity associated with combustion problems is the presence of regions in which the gradients are extremely large and the transients are very fast. Due to the exponential dependence of chemical reaction rate upon temperature, bulk of the combustion process is confined to a narrow zone (flame region), where steep gradients of temperature and concentration occur. Tracking down the location of the flame region and accurately representing the solution gradients in that zone, pose considerable computational difficulties. A variety of assumptions and simplifications have been invoked in the past during theoretical modelling, to make the problem tractable and to obtain solutions of practical importance.

Traditionally, two distinct approaches have been adopted in practice for the calculation of flame structure. Analytical and semi-analytical asymptotic methods have been employed to study flames described by simple, one-dimensional steady models (Buckmaster and Ludford, 1982; Williams, 1985; Kapila and Matkowsky, 1980; Ludford and Peters, 1984; Oran et al., 1981).

The second approach involves numerical solution of the complex system of differential equations for the combustion problem, with few simplifications. Today, several powerful numerical methods are available for predicting flame structures accurately in multi-dimensional problems. Yet, the problems of ignition and flame propagation present severe challenges to the available numerical methods, because of the small time and space scales associated with these processes.

In recent years, adaptive grid generation techniques have been developed for the application of finite difference or finite element solution procedures to flame propagation problems (Benkhaldoun et al, 1988a,b; Bieterman and Babuska, 1986; Dwyer et al., 1980,1982; Dwyer and Sanders, 1983a; Giovangigli and Smooke, 1989; Larroutrou, 1985, 1986; Ramos, 1985, 1987a-b, 1990a-b; Smooke, 1986b; Davis and Flaherty, 1982; Dervieux and Larroutrou, 1990; Miller and Miller, 1981a). These authors have used the concept of distributing the mesh points in a systematic way, so as to capture the regions of "large variation" of the solution. Such resolution of large gradient regions is important for capturing the physical phenomena and also for improving the solution accuracy in the flame front zone. The adaptive grid techniques reduce computational requirements drastically for a given accuracy level and thus make the calculation efficient and feasible. In these methods, the location of grid points are determined so as to minimize some measure of error in the computed solution.

Another important feature of the combustion problems is that the flame front may be moving with respect to time. Thus, in order to capture the flame front, moving mesh methods are needed. For

finite difference applications, some authors (Rai and Anderson, 1981; Dwyer and Sanders, 1983) have employed the grid speed strategy for controlling the motion of grid points. In the finite element context, Miller and Miller (1981a,b) and Gelinass et al (1981a), have proposed the movement of grid point locations in such a way as to minimize the total residue of the governing equations. This moving mesh procedure has been applied for flame propagation problems in multi dimensions.

Benkhaldoun and coauthors (1988a), have proposed two different gridding procedures for solving steady laminar flames. In their work, the mesh enrichment depends on the reaction rate and the adaption is carried out directionally. Also, in a later work, Benkhaldoun and Larrousturou (1988a, 1989) have developed a dynamic mesh generation procedure based on adaptive refinement and derefinement for the propagation of a premixed laminar flame. They have used unstructured finite element triangulation and the local refinement depends on the second order derivative of reaction rate and the diffusive flux.

In another paper, Benkhaldoun and Larrousturou (1987) have combined some features of the finite element and finite difference methods for studying the classical thermal diffusion model of wrinkled flame front in a gaseous mixture. They have used two different adaptive schemes, namely: (i) the 'dynamic rezoning' performed at each time step which allows the nodal locations to vary smoothly with time, (ii) the 'static rezoning' which is performed only at a few time levels during the calculation in order to redistribute the mesh points optimally. All the mesh points move with the same speed as the flame. The grid velocity is

chosen in such a way that the integral of each of the unknown variables on the computational domain remains constant or equivalently, it is given by the integral of the reaction rate.

Dervieux et al. (1989b) have reviewed various numerical methods for grid adaption in flame propagation problems. Different 1-D equi-distribution techniques and grid speed methods are described. Ramos (1983, 1985, 1990a-b) has used the temperature gradient for grid adaption while studying flame propagation by the finite element method. In a later work, the author (Ramos, 1991) has described adaptive finite element methods based on equi-distribution, elliptic grid generation and variational formulation.

In the present work, adaptive grid generation techniques which involve a *posteriori* error measures defined in terms of the computed finite element solution are employed. These techniques are applied for the study of one-dimensional premixed flame propagation and the performance of each technique is assessed by a detailed analysis of the error behaviour, solution accuracy and the adaptive mesh features.

3.2.2. Formulation of flame propagation problem :

The problem considered here is the simplified thermo-diffusive model which describes the laminar premixed flame propagation in one dimension (Fig. 3.1). A common situation where this problem finds application is the combustion of the air-gasoline mixture inside a Spark Ignition (SI) engine. In the SI engine, ignition is started with the help of the spark plug and a flame front then sweeps through the entire combustible mixture

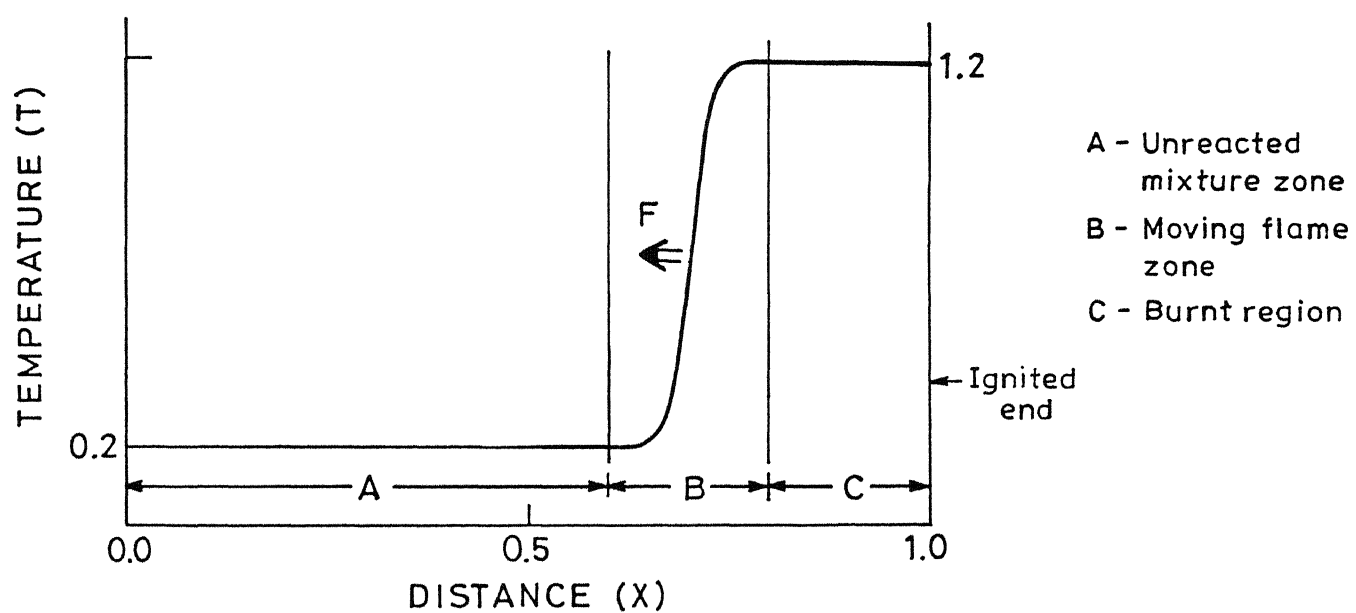


Fig. 3.1 A SCHEMATIC DIAGRAM OF ONE-DIMENSIONAL FLAME PROPAGATION .

(see, Fig. 3.2). The propagation of the flame is essentially one-dimensional, from one end of the combustion chamber to the other. It can be represented graphically in a manner as shown in Fig. 3.1.

The chosen model problem considers the the exponential temperature dependence of Arrhenius-type reactions and the diffusional characteristics of more general unsteady combustion problems. A channel of unit length, is filled with a cold mixture of the reacting species (fuel + oxidizer) in the right proportions. A one-step exothermic chemical reaction of the form $A \rightarrow B$ is assumed, where A represents the reactant mixture and B represents the combustion products. At time $t = 0$, the mixture is ignited at one end ($x = 1.0$) by supplying sufficient heat. Each calculation is begun by applying a linearly increasing temperature at the right-hand boundary (Fig. 3.1). As heat diffuses into the mixture, the rate of chemical reaction increases rapidly. The result is an acceleration of the reaction zone (flame) which moves away from the wall. After the ignition process is completed, a self-sustaining flame front develops and propagates into the unreacted mixture, until all of material A is converted into B. The speed at which the reaction zone moves (flame speed) will generally reach a constant value which will be maintained until the flame approaches the left side of the channel. This period of constant flame speed offers a convenient opportunity for assessing the accuracy of various numerical techniques.

In the present work, the above- described flame propagation model has been used for all the adaptive grid studies. Two specific cases, one corresponding to a stationary mixture and the

other dealing with flowing mixture, have been considered. The model is described by a stiff system of two coupled nonlinear reaction diffusion equations of the form

$$\left. \begin{aligned} \frac{\partial T}{\partial t} - \frac{\partial^2 T}{\partial x^2} + v \frac{\partial T}{\partial x} &= f(\rho, T) \\ \frac{\partial \rho}{\partial t} - \frac{\partial^2 \rho}{\partial x^2} + v \frac{\partial \rho}{\partial x} &= -f(\rho, T) \end{aligned} \right\} \quad t > 0, x \in (0, 1) \quad (3.1a)$$

with the boundary conditions

$$\left. \begin{aligned} \frac{\partial \rho}{\partial x}(0, t) &= 0 = \frac{\partial T}{\partial x}(0, t) \\ \frac{\partial \rho}{\partial x}(1, t) &= 0, T(1, t) = g(t) \end{aligned} \right\} \quad t > 0 \quad (3.1b)$$

and the initial conditions

$$\rho(x, 0) = 1, T(x, 0) = 0.2, x \in [0, 1] \quad (3.1c)$$

respectively. Here, $f(\rho, T)$ and $g(t)$ represent the following :

$$f(\rho, T) = Q \rho \exp(-K/T) \quad (3.1d)$$

$$g(t) = \begin{cases} 0.2 + t/(2 \times 10^4) & t \leq 2 \times 10^4 \\ 1.2 & t \geq 2 \times 10^4 \end{cases} \quad (3.1e)$$

with $Q = 3.52 \times 10^6$ and $K = 4.0$. (3.1f)

In these equations, T ($= T(x, t)$) is the temperature, ρ ($= \rho(x, t)$) is the mass fraction (density) of the reactant mixture, v is the flow velocity and the right hand side term $f(\rho, T)$ is the rate of chemical reaction. All the variables in the problem including the spatial and time co-ordinates, have been made dimensionless by scaling with the characteristic value of each variable. In the eq. (3.1d), K is the dimensionless activation energy and Q is the dimensionless pre exponential factor of the reaction rate. The

chosen numerical values of Q and K are typical of a combustion reaction.

These governing equations expressed through a coupled non-linear system are stiff due to the existence of a narrow flame zone. A few special cases of the general model are considered of interest here. If $V = 0$, the above problem simplifies to that of flame propagation in a stationary mixture which can be written as follows.

$$\left. \begin{aligned} \frac{\partial T}{\partial t} - \frac{\partial^2 T}{\partial x^2} &= f(\rho, T) \\ \frac{\partial \rho}{\partial t} - \frac{\partial^2 \rho}{\partial x^2} &= -f(\rho, T) \end{aligned} \right\} t > 0, x \in (0, 1) \quad (3.2)$$

with the same boundary and initial conditions as given by (3.1b-f). This problem has been studied in Chapter 4, while the general situation of flame propagation in a flowing mixture is analysed in Chapter 5. When $f(\rho, T)$ is replaced by a suitable $f(x, t)$, the resulting problem is linear. It can be solved analytically and the corresponding solution can be used for validating the numerical predictions. In the present study, such a linear problem exhibiting a wave front with appropriate boundary conditions has been termed as the "Test problem" while, the general non-linear reaction diffusion equations (3.1 a-f) are referred to as the "Flame problem". The test problem case has been used to validate the numerical solutions of problems considered in Chapter 4 and Chapter 5.

3.2.3. Salient features of the flame propagation problem :

In the reactive diffusive model presented above, the chemical

reaction rate exhibits a strong non-linear dependence with respect to temperature. In fact, according to the " Arrhenius kinetics", the rate of reaction depends exponentially upon the mixture temperature. One of the main difficulties in simulating flame propagation described by such reaction kinetics is the existence of disparate characteristic length scales. The stiffness of the problem appears essentially in the form of a narrow zone of sharp gradients around the flame front, dividing two regions in which the variables are almost constant. Thus, accurate representation of the inner structure of the thin flame plays a vital role in the overall simulation of the problem. Other aspects of interest in the study of the flame propagation problem are the predictions of the flame speed and the ignition transients. The critical step in the entire solution procedure is the accurate resolution of the flame front region at each time instant.

3.2.4. Computational requirements of flame propagation studies :

One of the major difficulties encountered during the simulation of moving front problems is the lack of information concerning the gradients of the dependent variable in different parts of the solution domain. Without this vital information, high gradient regions can not be resolved in a computationally efficient manner. Also, if a uniform grid is employed, a very large number of grid points will be needed to produce accurate results. However, truncation error on such a non-optimal grid will corrupt the computational results, because of the unnecessary usage of a small grid size in most of the solution domain. Besides, the requirement of obtaining a solution of given accuracy

with the minimum total cost and computational effort can not be met with the uniform grid. Thus, the chosen numerical method must be in a position to bunch grid points in high gradient regions and spread them out in regions where the solution varies weakly. It is expected to reduce the computational effort significantly compared to that for the uniform grid, thereby making the calculations efficient and feasible. These needs call for the use of a powerful adaptive grid procedure.

3.3. Mathematical formulation and finite element solution procedure :

The need for adaptive mesh generation during the solution of the front propagation problems, was amply elaborated in the previous sections. Adaptive grid refinement can be applied to the front propagation problems by employing various numerical techniques such as finite difference or finite element methods. The immediate aim of our work is application of adaptive finite element solution procedure to the problems as described earlier in which, the adaptive grid refinement is based on some computable error; measured in a properly chosen norm associated with the problem. This, in turn, requires the definition of a proper *a posteriori* measure of error so that the grid refinement is carried out at locations of large error, either in the solution or in the gradients of solution. For characterizing the predicted solution at each stage of grid adaption and for identifying the error measures, functional analysis tools have been used. In the ensuing sections, some important mathematical concepts based on functional analysis are presented which are useful for the grid

adaption as well as the finite element solution of flame propagation problems.

3.3.1. Mathematical Preliminaries :

In this section, we introduce some notations and definitions which will be used throughout this work. All functions treated here are real-valued functions. For each integer $j \geq 0$, $D^j f = \frac{\partial^j f}{\partial x^j}$.

Let Ω be a domain of variation of x independent of time, such that $0 < x < 1$. On the space of square integrable functions on Ω , the L^2 -inner product and norm are denoted by

$$\langle u, v \rangle = \int_{\Omega} uv \, dx$$

and,

$$\| u \|^2 = \langle u, u \rangle$$

respectively. The support of a function f is defined as

$$\text{Supp } f = \overline{\{ x : f(x) \neq 0 \}}.$$

Let $\bar{\Omega}$ be the closure of Ω ; $\partial\Omega$ be the boundary of Ω and $\bar{\partial\Omega} = \partial\Omega^- \cup \partial\Omega^+$. Also, $\bar{\Omega} = \Omega \cup \partial\Omega$. For any non-negative integer α , $C^\alpha(\Omega)$ denotes the space of functions with continuous derivatives up to and including order α in Ω . As usual, $C_0^m(\Omega)$ denotes all $C^m(\Omega)$ functions having compact support in Ω , while $C_0^\infty(\Omega)$ is the space of all infinitely differentiable functions with compact support in Ω .

For each non-negative integer m , the Sobolov space of order m on Ω denoted by $H_0^m(\Omega)$ is defined as the closure of $C_0^m(\Omega)$ with respect to the norm

$$\| u \|_{H_0^m(\Omega)}^2 = \sum_{j=0}^m \int_{\Omega} |D^j u(x, t)|^2 dx.$$

Equivalently, $H^m(\Omega)$ is the space of all functions whose distributional derivatives of order less than or equal to m , are in $L^2(\Omega)$. Clearly $L^2(\Omega) = H^0(\Omega)$. $H^m(\Omega)$ is a Hilbert space with the inner product defined by

$$\langle u, v \rangle_{H^m(\Omega)} = \sum_{j=0}^m \int_{\Omega} D^j u D^j v \, dx$$

For convenience sake, we use the notation

$$\phi_x = \frac{\partial \phi}{\partial x}, \quad \phi_{xx} = \frac{\partial^2 \phi}{\partial x^2} \quad \text{and} \quad \phi_t = \frac{\partial \phi}{\partial t}$$

We shall also employ the convention that the variables i, j, l, m, n, j^* and l^* take integer values. The total number of partial differential equations under consideration is denoted by NPDE. On $\bar{\Omega}$, we consider the family of partitions \mathcal{P}

$$I = \Delta(t) = \left\{ 0 = X_0 < X_1 < \dots < X_{l^*-1} < X_{l^*} = 1 \right\} \in \mathcal{P}$$

where $\Delta(t)$ is a space grid of Ω . The following symbols are used to denote an element (Ω_j) , the boundary of space domain $(\partial\Omega)$, step length (h_j) , maximum step length (\bar{h}) and minimum step length (\underline{h}) .

$$\Omega_j = \Omega_j(\Delta) = \left\{ x \in \mathbb{R}^1 : x_j < x < x_{j+1} \right\}; \quad j = 0, 1, \dots, l^*-1$$

$$\partial\Omega = \partial\Omega^- \cup \partial\Omega^+ \equiv \{0, 1\} \quad h_j = h_j(\Delta) = x_{j+1} - x_j;$$

$$\bar{h} = \bar{h}(\Delta) = \max_{0 \leq j \leq l^*-1} h_j(\Delta) \quad \underline{h} = \underline{h}(\Delta) = \min_{0 \leq j \leq l^*-1} h_j(\Delta)$$

\mathcal{P} is said to be a $K(\lambda)$ -regular family if constants $\kappa \geq 1$ and $\lambda > 0$ such that $\underline{h}(\Delta) \geq \lambda h_j^k(\Delta)$ for $j = 0, 1, \dots, l^*-1$. For any vector $y \in \mathbb{R}^1$ with components $y_0, y_1, y_2, \dots, y_{l^*}$, define l_∞ norm by

$$\|y\|_\infty = \max_{0 \leq i \leq l^*} (y_i)$$

3.3.2. Basic formulation in a general set up :

The reaction diffusion equations (3.2, 3.1b -3.1f) for the flame propagation in a stationary mixture can be incorporated into a general system of non-linear parabolic problems with the appropriate boundary conditions. In this section, we describe the finite element solution procedure for these equations which involves weak formulation, Galerkin procedure and time discretization scheme.

(a). Description of the general problem :

The reaction diffusion equations (3.2, and 3.1b-3.1f) for the flame propagation in a stationary mixture can be written in the more general form :

$$(u_1)_t - \left[a_1(x) (u_1)_x \right]_x = f_1(x, t, u) \quad (x, t) \in \Omega \times (0, t_f],$$

$$1 = 1, \dots, \text{NPDE}$$
(3.3a)

with the boundary conditions

$$\alpha_1(x) u_1 + \beta_1(x) (u_1)_x = g_1(x, t) \quad (x, t) \in \partial\Omega \times (0, t_f],$$

$$1 = 1, \dots, \text{NPDE}$$
(3.3b)

and initial conditions

$$u_1(x, 0) = \bar{u}_1^0(x), \quad x \in \Omega \quad (x, t) \in \Omega \times (0, t_f],$$

$$1 = 1, \dots, \text{NPDE}$$
(3.3c)

Here u ($= u(x, t)$) is a vector which consists of entries

$$u_1(x, t), u_2(x, t), \dots, u_{\text{NPDE}}(x, t).$$

For the flame propagation studies in the present work, the value of NPDE is 2. Further, the solutions $u_1(x, t)$ and $u_2(x, t)$ represent the mixture temperature and density respectively. In

the following discussion, we may refer to the solution vector by its component u_1 or simply by u . Also, it is to be noted that u appears as a vector in the non-linear reaction term $f_1(x,t,u)$ of the above equation. The coefficients $\alpha_1(x)$ and $\beta_1(x)$ determine the type of boundary conditions and $u_1^0(x)$ is the initial condition at time $t = 0$ for each u_1 . For $x \in \bar{\Omega}$, we consider $a_1(x) \geq 0$, with $a_1(x) > a_0 > 0$ for atleast one index 1 . Further, it is assumed that $\beta_1(\partial\Omega^-) \leq 0$; $\beta_1(\partial\Omega^+) \geq 0$ and $\alpha_1(x) \geq 0$ for $x \in \partial\Omega$, $1 = 1, \dots, \text{NPDE}$. We also assume that with reasonable problem data, the solution $u_1(x,t)$ of the system of equations (3.2, 3.1b-3.1f) exists and a smooth mapping of $[0,1]$ into the Hilbert space

$$H_1^{\text{NPDE}}(\Omega) = \left\{ v = \{v_1\}_{1=1, \text{NPDE}} : v_1, (v_1)_x \in L_2(\Omega); 1 = 1, \dots, \text{NPDE} \right\}.$$

Let us define the seminorm

$$\|v\| = \left\{ \sum_{1=1}^{\text{NPDE}} \langle a_1(v_1)_x (v_1)_x \rangle \right\}^{1/2}; \quad v \in H_1^{\text{NPDE}}$$

which is a norm on $H_1^{\text{NPDE}}(\Omega)$ provided appropriate conditions on a_1, α_1 , and β_1 ($1 \leq 1 \leq \text{NPDE}$) hold.

(b). Weak formulation :

Now the weak solution $u_1(x,t)$ ($= u_1$) of (3.3a-c) is defined as follows : Find a function

$$\tilde{u}_1 : [0, t_f] \longrightarrow H^1(\Omega) \quad 1 = 1, \dots, \text{NPDE}.$$

such that

$$\alpha_1 u_1(x,t) + \beta_1(x) (u_1)_x = g_1(x,t) \quad \text{for } 1 = 1, \dots, \text{NPDE} \quad \text{on } x \in \partial\Omega$$

is satisfied and $\tilde{u}_1(x,t)$ satisfies the equations

$$\begin{aligned}
& \left\{ \left\langle \frac{\partial}{\partial t} \tilde{u}_1(\cdot, t), v_1 \right\rangle + \left\langle a_1 \frac{\partial}{\partial x} \tilde{u}_1(x, t), \frac{\partial}{\partial x} v_1 \right\rangle \right. \\
& \quad \left. + a_1(x) \frac{\alpha_1(x)}{\beta_1(x)} \tilde{u}_1(x, t) v_1 \right|_{\partial\Omega^-}^{\partial\Omega^+} \Bigg\} \\
& = \left\langle f_1(x, t, \tilde{u}), v_1 \right\rangle + \frac{\alpha_1(x)}{\beta_1(x)} g_1(x, t) v_1 \Big|_{\partial\Omega^-}^{\partial\Omega^+} \\
& \quad t \in (0, t_f], \quad i = 1, \dots, \text{NPDE}. \quad (3.4a)
\end{aligned}$$

$$\text{and } \langle \tilde{u}_1(x, 0), v_1 \rangle = \langle \tilde{u}_1^0(x), v_1 \rangle, \quad t \in (0, t_f], \quad i = 1, \dots, \text{NPDE}. \quad (3.4b)$$

for all $v_1 \in H^1(\Omega)$. In fact (3.3 a-c) to (3.4 a-b) are equivalent if $\tilde{u}_1(x, t)$ has sufficient regularity. Let us now turn to the formulation of Galerkin method in space variable called the semidiscrete approximation for the solution

$$\tilde{u}(x, t) = \left(\tilde{u}_1(x, t), \tilde{u}_2(x, t), \dots, \tilde{u}_{\text{NPDE}}(x, t) \right)$$

of the equations (3.4 a-c).

(c). Semi discrete approximation :

We describe the construction of the finite dimensional solution space in which the approximation for the solution $\tilde{u}_1(x, t)$ in H_p^1 ($1 < p < \infty$) lies. Generally the finite dimensional solution space, considered to approximate $\tilde{u}_1(x, t)$ in the Galerkin procedure, belongs to a regular family and satisfies certain inverse properties. In the finite element literature, such a class of finite dimensional sub spaces which depend on the mesh size (h) is referred to as $S_h^{r,k}$ family. We recall here the definitions of $S_h^{r,k}$ families as described in Babuska (1972) and Oden and Reddy (1976).

Construction of finite dimensional solution space :

Let $r \geq 1$ be a given integer and $\mathbf{k} = (k_0, k_1, \dots, k_{l^*-1})$ be a vector of integers corresponding to the l^* sub-intervals Ω_j such that

$$k_j \geq 2r, \quad j = 0, 1, 2, \dots, l^*-1$$

Then $S_h^{r,\mathbf{k}}$ ($= S_h^{r,\mathbf{k}}(\Delta)$) is defined as the space of $(r-1)$ times continuously differentiable functions on Ω for which the restriction to Ω_j ($j = 0, 1, 2, \dots, l^*-1$) is a polynomial of degree k_j-1 (Ciarlet, 1978). In fact, for any $\tilde{u}_1 \in S_h^{r,\mathbf{k}}(\Delta)$, let $(\tilde{u}_1)_j$ be the restriction of \tilde{u}_1 to Ω_j ; $j = 0, 1, 2, \dots, l^*-1$. Then for $k_j = 2r$, the polynomial $(\tilde{u}_1)_j$ is defined by its value and those of its $(r-1)$ first derivatives at the end points of Ω_j . For $k_j \geq 2r$, $(u_1)_j$ is a linear combination of polynomials for which the value and those of the $(r-1)$ first derivatives are prescribed at the end points of Ω_j and k_{j-2r} polynomials which together with their $(r-1)$ first derivatives vanish at both these end points. Clearly, we have $S_h^{r,\mathbf{k}}(\Delta) \subset H_p^r(\Omega)$ for any $1 < p < \infty$. Then a continuous time Galerkin approximation of the solution

$$U_1(x, t) = \left(U_1(x, t), U_2(x, t), \dots, U_{NPDE}(x, t) \right) \quad (3.5)$$

is defined by a function

$$U_1(x, t) : [0, t_f] \longrightarrow S_h^{r,\mathbf{k}}$$

such that

$$\left\langle \frac{\partial U_1^1(\cdot, t)}{\partial t}, V_1 \right\rangle + \left\langle a_1 \frac{\partial U_1(x, t)}{\partial x}, \frac{\partial V_1}{\partial x} \right\rangle = \left\langle \tilde{f}_1(x, t, U), V_1 \right\rangle$$

$$\text{for all } V_1 \in S_h^{r,\mathbf{k}}; t \in (0, t_f], \text{ and } i = 1, \dots, NPDE.$$

(3.6a)

where

$$S_h^{r,\mathbf{k}} = \left\{ g \in S_h^{r,\mathbf{k}}; g = 0 \text{ on } \partial\Omega \right\}$$

with suitably chosen initial conditions

$$\langle U_1(x,0), V_1 \rangle = \langle \hat{U}_1(x), V_1 \rangle, \quad t \in (0, t_f], \quad \text{and } i = 1, \dots, \text{NPDE}. \quad (3.6b)$$

In the above expressions, $\hat{U}_1(x) \in S_h^{r,k}(\Delta)$ is the L_2 projection of $U_1(x)$ in $S_h^{r,k}(\Delta)$ and $\tilde{f}_1(x,t,U(x,t))$ is an interpolation of $f_1(x,t,U(x,t))$.

Expression in terms of the basis :

Let $\psi_{1,j}(x) \in S_h^{r,k}$ for each $j = 1, 2, \dots, N_h$ be a basis for the finite dimensional solution space $S_h^{r,k}$ for each $i = 1, \dots, \text{NPDE}$ where $\dim(S_h^{r,k}) = N_h$. Then the Galerkin approximation $U_1(x,t)$ can be written as

$$U_1(x,t) = \sum_{j=1}^{N_h} U_{1,j}[t] \psi_{1,j}(x), \quad t \in (0, t_f], \quad i = 1, \dots, \text{NPDE}. \quad (3.7)$$

Here, the vector $U_1[t] = \left\{ U_{1,j}[t] \right\}_{j=1}^{N_h}$ represents the solution vector for each $i = 1, \dots, \text{NPDE}$ and in the flame problem this vector is equivalent to temperature and density at each nodal value of the mesh Δ . Substituting the above expression of $U_1(x,t)$ and taking

$$U_{1,j}(x,t) = V_{1,j}(x,t) \quad \text{for } i = 1, \dots, \text{NPDE}$$

in the equation (3.6a-b), we obtain an initial value problem for the system of N_h ordinary differential equations for each partial differential equation. Define

$$M_1 = [(M_1)_{ij}] \text{ as the matrix with entries } (M_1)_{ij} = \langle \psi_{1i}, \psi_{1j} \rangle$$

$$K_1 = [(K_1)_{ij}] \text{ as the matrix with entries } (K_1)_{ij} = \langle a_1 \nabla \psi_{1i}, \nabla \psi_{1j} \rangle$$

$F_1 = \{F_{1,j}\}$ is the right-hand side vector with entries

$$F_{1,j} = \langle \tilde{f}_1(x, t, U[t]), \psi_{1,j} \rangle$$

$\gamma_1 = \{\gamma_{1,j}\}$ as the vector with entries $\gamma_{1,j} = U_{1,j}[0]$, which are the components of initial approximation $\hat{U}_1(x, 0)$ and the dimension of all these quantities equal to N_h , the dimension of $S_h^{r,k}(\Delta)$. Also, for all these variables, the subscript '1' varies from 1 to NPDE. Employing the above matrix notation, equation (3.6a-b) can be written in an equivalent form as :

$$\begin{bmatrix} M_1 \end{bmatrix} \left\{ \frac{dU_1[t]}{dt} \right\} + \begin{bmatrix} K_1 \end{bmatrix} \{U_1[t]\} = \{F_1(U[t])\} \quad \text{for } i=1, \dots, \text{NPDE} \quad (3.8a)$$

with the initial conditions

$$\{U_1[0]\} = \{\gamma_1\} \quad \text{for } i = 1, \dots, \text{NPDE} \quad (3.8b)$$

The matrix $\begin{bmatrix} M_1 \end{bmatrix}$ is positive definite $\forall t \geq 0$ and the ODE system has a unique local solution for $t \geq 0$ and $i = 1, \dots, \text{NPDE}$.

3.3.3. Method of solution :

The system of ordinary differential equations (3.8a-b) in

$$U_{1,j}[t] \quad j = 1, \dots, N_h^1; \quad i = 1, \dots, \text{NPDE}$$

are nonlinear as $F_1(U[t])$ is nonlinear in $U[t]$. Thus, the determination of $U_1(x, t)$ amounts to solving the system of N_h differential equations in $U_{1,j}[t]$ with appropriate initial conditions. These equations can be solved by a variety of numerical methods available for the solution of a system of ODE, one of which is to discretize in time and generate a set of simultaneous algebraic equations. Several researchers have used

the integration packages which are designed to solve specific classes of ordinary differential equations accurately and efficiently (see Hindmarsh and Gear, 1979; Axelsson, 1976; Bieterman and Babuska, 1986; Benkhaldoun and Larrouturou, 1987). Many ideas of these solvers have been utilized in the context of evolution problems for PDE's and particularly in conjunction with finite element methods. As the resulting difference equations are nonlinear, some authors (Davis and Flaherty, 1982; Ramos, 1985) have adopted the incremental Newton-Raphson method. It is well known that the discrete representation of the ODE operators have certain properties that lead to restrictions on time step and the element length, to reduce phase errors which generate oscillations. The choice of a particular scheme for solving these ODE systems, therefore, depends upon the step size restrictions and the accuracy of the method.

In the present study, the Crank-Nicolson-Galerkin method has been employed to discretize the problem with respect to the time variable. This procedure is second order accurate in terms of the time step Δt . The resulting non-linear algebraic equations are solved by an iterative method. The iterative method converges free of oscillations if adequately small time steps Δt (of order 10^{-5} to 10^{-6}) are chosen.

(a). Time discretization :

Let Δt be the time step and $U_1^i(.,t)$ be the approximate solution in $S_h^{r,k}(\Delta^i)$ of $U_1(.,t)$ at $t = t_i = i \Delta t$ ($1 \leq i \leq NPDE$). Here, the finite dimensional solution space $S_h^{r,k}(\Delta^i)$ is constructed at the time t_i of the mesh Δ with $\dim(S_h^{r,k}(\Delta^i)) = N_h^i$

The semi-discrete equation is then discretized with respect to the time variable in a symmetric fashion around the point $t_{i-1/2} = (i-1/2)\Delta t$ producing a second order accurate method in time. More precisely, we define $U_1^i(.,t)$ in $S_h^{r,k}(\Delta^i)$ recursively by

$$\begin{aligned} & \left\langle \frac{\partial}{\partial t} U_1^{i-1/2}(.,t), \psi_1 \right\rangle + \left\langle a_1 \frac{\partial}{\partial x} \left(\frac{U_1^i(.,t) + U_1^{i-1}(.,t)}{2} \right), \frac{\partial}{\partial x} \psi_1 \right\rangle \\ &= \frac{1}{2} \left\langle \tilde{f}_1^i(.,t, U_1^i(.,t)) + \tilde{f}_1^{i-1}(.,t, U_1^{i-1}(.,t)), \psi_1 \right\rangle \end{aligned} \quad (3.9)$$

for ψ_1 in $S^{r,k}(\Delta^i)$; $t \in [t_{i-1}, t_i]$, $i = 1, \dots, NPDE$;

with appropriately chosen boundary conditions. The expression for the term $\frac{\partial}{\partial t} U_1^{i-1/2}(.,t)$ can be derived as follows:

$$U_1^i(.,t) = U_1^{i-1/2}(.,t) + \frac{\Delta t}{2} \frac{\partial}{\partial t} U_1^{i-1/2}(.,t) + O((\Delta t)^2) \quad (3.10a)$$

$$U_1^{i-1}(.,t) = U_1^{i-1/2}(.,t) - \frac{\Delta t}{2} \frac{\partial}{\partial t} U_1^{i-1/2}(.,t) + O((\Delta t)^2) \quad (3.10b)$$

$$U_1^{i-1/2}(.,t) = U_1^{i-1}(.,t) + \frac{\Delta t}{2} \frac{\partial}{\partial t} U_1^{i-1}(.,t) + O((\Delta t)^2) \quad (3.10c)$$

$$U_1^{i-1/2}(.,t) = U_1^i(.,t) - \frac{\Delta t}{2} \frac{\partial}{\partial t} U_1^i(.,t) + O((\Delta t)^2) \quad (3.10d)$$

From these equations, one can write

$$\frac{\partial}{\partial t} U_1^{i-1/2}(.,t) = \frac{\partial}{\partial t} \left(\frac{U_1^i(.,t) + U_1^{i-1}(.,t)}{2} \right) = \frac{1}{\Delta t} \left(U_1^i(.,t) - U_1^{i-1}(.,t) \right) \quad (3.11)$$

Thus the final discretized equations can be written as

$$\left\langle U_1^i(.,t) - U_1^{i-1}(.,t), \psi_1 \right\rangle + \Delta t \left\langle a_1 \frac{\partial}{\partial x} \left(\frac{U_1^i(.,t) + U_1^{i-1}(.,t)}{2} \right), \frac{\partial}{\partial x} \psi_1 \right\rangle$$

$$\begin{aligned}
&= \frac{\Delta t}{2} \left\langle \tilde{f}_1^1(\cdot, t, U^1(\cdot, t)) + \tilde{f}_1^{1-1}(\cdot, t, U^{1-1}(\cdot, t)), \psi_1 \right\rangle \\
&\quad \text{for } \psi_1 \text{ in } S^{r,k}(\Delta^1); \quad i = 1, \dots, \text{NPDE}; \quad t \in [t_{i-1}, t_i]
\end{aligned} \tag{3.12}$$

It is noted that in the above system of equations (3.11) the boundary conditions have been incorporated in the choice of $\psi_{1,j}$'s. Thus the matrix equations for the time discretized form (3.12) can now be written as follows :

$$\begin{aligned}
&\left[M_1 + \frac{\Delta t}{2} K_1 \right] \left\{ U_1^1[t] \right\} - \left[M_1 - \frac{\Delta t}{2} K_1 \right] \left\{ U_1^{1-1}[t] \right\} \\
&= \frac{\Delta t}{2} \left\{ F_1^1(U^1[t]) \right\} + \frac{\Delta t}{2} \left\{ F_1^{1-1}(U^{1-1}[t]) \right\} \\
&\quad t \in [t_{i-1}, t_i]; \quad i = 1, \dots, \text{NPDE}
\end{aligned} \tag{3.13}$$

$$\text{where } U_1^1[t] = \sum_{j=1}^{N_h^1} U_{1,j}^1[t] \psi_{1,j}(x); \quad t \in [t_{i-1}, t_i]; \quad i = 1, \dots, \text{NPDE}$$

Since the solution vector, namely,

$$\left\{ U_1^{1-1}[t] \right\} \quad \text{for each } i = 1, \dots, \text{NPDE};$$

is known, the second term on both sides of the equation (3.13) is known. Thus the matrix system can be rewritten as given below for each $t \in [t_{i-1}, t_i]$.

$$\begin{aligned}
&\left[A_1^i \right] \left\{ U_1^1[t] \right\} = \left\{ G_1^1(U^1[t]) \right\} + \left\{ H_1^{1-1}(U^{1-1}[t]) \right\} \\
&\quad i = 1, \dots, \text{NPDE}; \quad t \in [t_{i-1}, t_i]
\end{aligned} \tag{3.14}$$

where

$$\begin{aligned} \left[A_1^i \right] &= \left[M_1 + \frac{\Delta t}{2} K_1 \right] \\ \left\{ G_1^i(U^i[t]) \right\} &= \frac{\Delta t}{2} \left\{ F_1^i(U^i[t]) \right\} \\ \left\{ H_1^{i-1}(U^{i-1}[t]) \right\} &= \left[M_1 - \frac{\Delta t}{2} K_1 \right] \left\{ U_1^{i-1}[t] \right\} + \frac{\Delta t}{2} \left\{ F_1^{i-1}(U^{i-1}[t]) \right\} \end{aligned}$$

(b). Iterative method :

The above equations (3.14) can be solved by the following iterative method. The main aim is to obtain the finite element solution at $t = t_i$ by solving the matrix system of equations (3.14) with the help of the finite element solution available at $t = t_{i-1}$. Since the unknown variable appears in both sides of the equations (3.14), the corresponding equations are nonlinear. A proper initial guess solution is very important in order to obtain the convergence of iterative method. For the starting iteration, the initial guess solution for the evaluation of source vector at $t = t_i$ is provided by the finite element solution available or computed at previous $t = t_{i-1}$. But, at the subsequent iterations, the freshly computed solution at $t = t_i$ is used in the source vector at $t = t_i$. Now, the right hand side of the matrix equations (3.14) is evaluated and then the matrix system (3.14) is solved by direct methods. Finally, the iterative procedure is repeated and stopped after the maximum difference of successive solution vectors becomes less than a given tolerance (ϵ) which is of the order (10^{-6}). The convergence of the iterative method poses no problems since small time steps have been considered.

This iterative procedure is explained by the following steps of the algorithm. In the algorithm, we denote $\hat{U}[t]$ as the starting guess solution for the iterative method, $\bar{U}[t]$ the computed solution, \bar{M} the total number of iterations performed and $\left(\hat{U}_1^i[t] \right)^{\bar{M}}$ as the solution at \bar{M}^{th} iteration respectively.

Step :1 Choose the initial guess solution at $t = t_1$ ($1 \leq i \leq NPDE$).

i.e., choose $\left(\hat{U}_1^i[t] \right)^{\bar{M}} = \hat{U}_1^i[t] = \bar{U}_1^{i-1}[t]$ ($1 \leq i \leq NPDE$)

where \bar{M} is the iteration count set equal to 0 initially.

Step :2 Increase the iterative counter \bar{M} by 1.

Step :3 Calculate the right hand side vectors at $t = t_1$

i.e., $\left\{ G_1^i \left(\hat{U}_1^i[t] \right)^{\bar{M}} \right\}$ at $t=t_1$ and $\left\{ H_1^{i-1} \left(\bar{U}_1^{i-1}[t] \right) \right\}$ at $t=t_{i-1}$

step :4 Solve the matrix system (3.14) for each i ($1 \leq i \leq NPDE$)

and denote the solution by $\bar{U}_1^i[t] = \left(\hat{U}_1^i[t] \right)^{\bar{M}+1}$ at $t = t_1$

Step :5 Convergence check

If $\left\| \left(\hat{U}_1^i[t] \right)^{\bar{M}} - \left(\hat{U}_1^i[t] \right)^{\bar{M}+1} \right\|_{\infty} \leq \epsilon$ for each i , ($1 \leq i \leq NPDE$)

then accept $\left(\hat{U}_1^i[t] \right)^{\bar{M}} = \bar{U}_1^i[t]$ as the computed solution at $t = t_1$ and stop the iterative method.

Step :6 Perform iterations

If the convergence criterion (step 5) fails then

replace $\left(\hat{U}_1^i[t] \right)^{\bar{M}}$ by $\left(\hat{U}_1^i[t] \right)^{\bar{M}+1}$ and go to step 2.

3.4. Description of error measures used for grid adaption in the finite element method :

In this section, we describe the various error measures used

for the grid adaption in the literature.

3.4.1. History of a posteriori error estimates used for grid adaption :

Several researchers (Babuska and Rheinboldt, 1978a, 1979b, 1979c, 1981a, 1982a; Bank, 1983, 1986; Bieterman and Babuska, 1982a, 1982b; Carey and Humphery, 1981; Eriksson and Johnson 1987, 1988; Gago et al., 1983; Oden et al., 1986a, 1986c; Oden and Demkowicz, 1986b; Szymczak and Babuska, 1984; Zhu and Zienkiewicz, 1991) have used the *a posteriori* error estimates based on residuals as a basis for the local enrichment of the solution. The analysis of error estimates and refinement strategies have been greatly aided by the the recent developments in the mathematical theory of finite element method. For this purpose the concept of error estimators and indicators have been developed, the estimators giving a relatively accurate measure of the error in a given mesh and the indicators a measure of change in the energy associated with the introduction of new variables. These *a posteriori* error estimates provide a basis for the design of adaptive finite element solvers.

For finite element grid adaptations, the *h* and *h-p* refinement techniques have been widely used by several authors (Babuska and Rheinboldt, 1982a; Babuska et al., 1984; Demkowicz et al., 1989; Oden and Demkowicz, 1986c; Oden et al., 1989; Szabo, 1986; Gago et al., 1983, Zienkiewicz et al., 1989) in which the convergence properties which can be improved, there by reducing the error in the finite element solution. The practical implementation of the optimal *h-p* refinement strategy is of difficult and has not yet

dense meshing and high p degree approximation procedures which are not simply adapted to practice. Most of the general ideas on the error based refinement strategies for finite element applications have been presented in the survey paper by Babuska and Rheinboldt, 1982a; Oden and Demkowicz, 1986c.

Babuska and Rheinboldt (1978a, 1981a) have derived computable *a posteriori* error estimates for finite element solutions in an asymptotic form as $h \rightarrow 0$ where, h measures the size of the elements and the grid refinement depends upon the *a posteriori* error estimate. Further, the same authors have also derived the error estimates in order to achieve an optimal mesh in which all the error indicators are nearly equal.

These ideas have been extended to parabolic problems involving a linear self adjoint operator of the second order by several authors (Bieterman and Babuska, 1982a,b; Eriksson and Johnson, 1988; Lohner, 1987; Oden et al., 1986a). Bieterman and Babuska (1982a,b) have derived computable *a posteriori* error estimates of the space discretization error in the finite element method. The effectiveness of the error estimation is related to the regularity of the solution, mesh family type and asymptotic range for the mesh size. The theory presented in their work, provides the basis for the analysis and adaptive construction of time dependent meshes in spatial region. With the help of such spatial error and discretization error estimates, the grid spacing can be selected so as to equidistribute the local error and, grid refinement can be continued until the total error meets a certain tolerance criterion.

The availability of precisely defined reliable error

estimators appears to be central to the design of adaptive procedure, but for the nonlinear problems like flame problem, the error estimates have not yet been developed although such estimates are important from the computational point of view.

3.4.2. Description of a posteriori error estimates

In this section, we describe the important error measures for time dependent problems which are useful in the present work.

(a). Description of the model problem :

Following Bieterman and Babuska (1982a, 1982b) we describe the following model problem and the various error measures used for the linear parabolic problems. Consider the following model problem :

$$\frac{\partial}{\partial t} u(x,t) - \frac{\partial}{\partial x} \left[a(x) \frac{\partial}{\partial x} (u(x,t)) \right] + b(x)u(x,t) = f(x,t) \\ (x,t) \in I \times (0, t_f) \quad (3.15a)$$

with the boundary conditions

$$u(0,t) = 0 = u(1,t); \quad t \in (0, t_f) \quad (3.15b)$$

and initial conditions

$$u(0,x) = u_0(x); \quad x \in I \quad (3.15c)$$

Here, $I = (0,1)$ is an unit open interval in \mathbb{R}^1 . Further, let $a(x)$ and $b(x)$ be sufficiently smooth functions on I for which

$$0 \leq \underline{a} \leq a(x) \leq \bar{a} < \infty; \quad 0 \leq b(x) \leq \bar{b} < \infty \quad \forall x \in \bar{I}$$

Let \mathcal{P} be a family of mesh partitions of I and

$$\Delta = \left\{ 0 = x_0 < x_1 < \dots < x_{N(\Delta)} = 1 \right\} \in \mathcal{P}$$

where $N(\Delta)$ is the dimension of corresponding finite dimensional solution space in which the solution lies. Let $h_j = x_{j+1} - x_j$ and $\Omega_j = [x_j, x_{j+1}]$. With reasonably smooth data for u_0 and f , we

assume that the solution of equations (3.15a-c) satisfying certain regularity properties. The details of the regularity class can be seen in the references (Babuska and Rheinboldt, 1978; Bieterman and Babuska, 1982a,b).

Now for each $\Delta \in \mathcal{P}$, the finite element approximation is obtained in a routine way as explained in the section (3.3.3). Let $u(x,t)$ be the exact solution and $\bar{U}(x,t)$ be the computed solution of the system of equations (3.15a-c) respectively.

The *a posteriori* error estimates of the elliptic problem play a crucial role for the development of the necessary theory for the parabolic problems. Here, the model problem for the elliptic case (see Babuska and Rheinboldt, 1978a) is deduced by dropping the time derivative term in the equation (3.15a) given by

$$-\frac{\partial}{\partial x} \left[a(x) \frac{\partial}{\partial x} u(x) \right] + b(x)u(x) = f(x); \quad x \in I \quad (3.16)$$

with the appropriate boundary conditions.

(b). *h* refinement technique :

A posteriori error estimate for elliptic problems :

The following asymptotic *a posteriori* error estimate for finite element solutions has been derived by Babuska and Rheinboldt (1978a) for a class of linear elliptic equations of type (3.16).

$$\|e(x)\|_E \leq c \left(\sum_{j=0}^{N(\Delta)-1} \eta_j^2 \right)^{1/2} (1+O(h)) \text{ as } h \rightarrow 0 \quad (3.17)$$

where, $e(x) = u(x) - \bar{U}(x)$, u , \bar{U} being the exact and computed solution of equation (3.16) and the energy norm of the exact

error ($\|e\|_E$) is given by

$$\|e(x)\|_E^2 = \left\langle a(x) \frac{\partial e(x)}{\partial x}, \frac{\partial e(x)}{\partial x} \right\rangle + \left\langle b(x) e(x), e(x) \right\rangle \quad (3.18)$$

Here, c is a constant which depends upon the degree of approximation in the finite element method. In the equation (3.16), the computable quantity η_j is called the error indicator and is expressed by :

$$\eta_j^2 = \frac{h_j^2}{\bar{a} \pi^2} \int_{x_j}^{x_{j+1}} r_j^2 dx, \quad j = 0, 1, 2, \dots, N(\Delta)-1 \quad (3.19)$$

where the elemental residual for the element Ω_j is given by

$$r_j = r_j(x) = - \frac{\partial}{\partial x} \left(a(x) \frac{\partial \bar{U}(x)}{\partial x} \right) + b(x) \bar{U}(x) - f(x),$$

$$x \in [x_j, x_{j+1}],$$

$$\bar{a} = a \left(\frac{1}{2} (x_j + x_{j+1}) \right),$$

and $N(\Delta)$ is the number of nodes in the domain. The *a posteriori* error estimate in the equation (3.17), namely

$$\left(\sum_{j=0}^{N(\Delta)-1} \eta_j^2 \right)^{1/2} \quad (3.20)$$

provides an easily computable bound for the total solution error under the energy norm. Further, the error indicator η_j provides a measure of the error based on the local elemental residue r_j of the j^{th} element of the mesh. The coefficient $1/\pi^2$ which can be replace by $1/12$ in the equation (3.19) has been obtained by an eigen value analysis of the associated operator satisfied by finite element residue of the approximate solution.

A posteriori error estimate for parabolic problems :

Using this error analysis for the elliptic problems,

Bieterman and Babuska (1982a,b) proposed an *a posteriori* error estimate for the time dependent spatial meshes and presented the theory for the estimation of energy norm of the space discretization error at each time. In order to assess the performance of the *a posteriori* error in actual practice the authors have used various notions such as relative error, true error and effectivity ratio. We define some of the error measures which are used in conjunction with the time dependent problems.

The energy norm of the exact error is defined by

$$\|e(\cdot, t)\|_E^2 = \left\langle a(x) \frac{\partial}{\partial x} e(x, t), \frac{\partial}{\partial x} e(x, t) \right\rangle + \left\langle b(x) e(x, t), e(x, t) \right\rangle$$

The gradient norm of the exact error is defined by

$$\| \| e(\cdot, t) \| \| = \left\{ \int_1 a(x) \left(\frac{\partial}{\partial x} (e(x, t)) \right)^2 dx \right\}^{1/2}$$

The *a posteriori* error estimator $\mathcal{E}(t)$ is defined by

$$\mathcal{E}(t) = \left\{ \sum_{j=1}^{N(\Delta)-1} \eta_j^2(t) \right\}^{1/2} \quad (3.20)$$

where the local error indicators $\eta_j^2(t)$ are defined by

$$\eta_j^2 = \frac{h_j^2}{12\bar{a}} \int_{x_j}^{x_{j+1}} r_j^2(x, t) dx \quad (3.21)$$

Here, the residual is given by the expression

$$r_j(x, t) = \frac{\partial}{\partial t} \bar{U}(x, t) - \frac{\partial}{\partial x} \left(a(x) \frac{\partial}{\partial x} \bar{U}(x) \right) + b(x) \bar{U}(x, t) - f(x, t) \\ (x, t) \in \Omega_j \times (0, t_f).$$

It is computable for each $t > 0$ on each Ω_j with the help of the computed solution $\bar{U}(x, t)$ of the equations (3.15a-c). The

Relative error $E_{REL}(t)$ is defined as the ratio of the computed gradient to the exact gradient in some properly chosen norm associated with the problem and is given by

$$E_{REL}(t) = \|e(.,t)\| / \|U(.,t)\| \quad (3.22)$$

Effectivity ratio $\theta(t)$ is given by

$$\theta(t) = \mathcal{E}(t) / \|e(.,t)\| \quad (3.23)$$

and it indicates relative growth of a *a posteriori* error and gradient norms of the error. Bieterman and Babuska (1982a,b) have shown the existence of asymptotic estimates with respect to spatial mesh size for the energy norm of the error ($\|e(.,t)\|_E$). The mathematical theory presented in their work provides the basis for the estimation of $\|e(.,t)\|_E$ of the Galerkin approximation of time dependent problems. Further, they have carried out an asymptotic analysis for the estimation of energy norm and have shown that for higher terms in the space mesh size

$$\|e(.,t)\|_E \sim \|e(.,t)\|$$

The latter error estimate ($\|e(.,t)\|$) of the above equation is related to the computed solution gradients. Since the exact solution is not available, the *a posteriori* error estimator $\mathcal{E}(t)$ has been used in place of $\|e(.,t)\|$ for the adaptive procedures. In order to assess the performance of $\mathcal{E}(.)$, which depends on local elemental error indicators, the concept of effectivity ratio $\theta(t)$ has been developed by these authors. Further, they have studied in detail the behaviour of $\mathcal{E}(.)$ and the relative error $E_{REL}(.)$. From the above definition of $\theta(t)$, it can be said that, $\mathcal{E}(t)$ is an effective estimator if there exists a reasonable constant 'C' (depending on the admissible solution class for Eqs.(3.15a-c) and

the admissible class of mesh partitions but not on the magnitude of the data U_0, f , and g) such that

$$1/\theta(t) \leq C \quad \forall \quad t > 0.$$

In practice, it is desirable that $\theta(t)$ be close to unity in order that the computed error is a proper measure of the true error. Therefore, as the partitions of $(0,1)$ are refined $\theta(t) \rightarrow 1$. The quality of the estimator $\mathcal{E}(t)$ was theoretically analyzed for the uncoupled partial differential equations. It is to be noted that relative error expression signifies how well the gradients are approximated. This is important for front propagation problems due to the presence of high gradient regions. The authors have also applied the h refinement technique for the reaction diffusion type of problems.

(c). h - p refinement:

Extending the work of Babuska and Rheinboldt (1978a), Kelly et al (1983) and Gago et al. (1983) have suggested an error estimator for higher order elements for linear elliptic equations of type (3.15). They have used the super-convergence results at nodes in their assumptions of exact computed values. (i.e. $e(x_i) = 0$ for each grid point $x_i \in I$). Here, the local error expression for the *a posteriori* error estimate is given by

$$\eta_j^2 = \frac{h_j^2}{12 \bar{a} p_j^2} \int_{x_j}^{x_{j+1}} r_j^2(x) dx \quad (3.24)$$

where p_j is the local degree of approximation used in the element Ω_j and $r_j(x)$ is the local elemental residue of the governing equations in Ω_j .

3.4.3. A posteriori error estimates used in the present work :

In the present work, we have used the error estimates proposed by the Babuska and co-researchers (Bieterman and Babuska, 1982a,b; Kelly et al., 1983; Gago et al., 1983) for the computational simulation of both the Test problem and the Flame propagation problem in a stationary mixture, by the finite element method. Further, the error estimates have been used for the h and h - p refinement techniques in Chapter 4. Since the flame propagation problem is non-linear and time dependent, no mathematical theory has been developed for the validation of numerical simulation of the present work. However, the *a posteriori* error estimates for both refinement methods have been tested using the analytical solution of the test problem. In addition to the error estimates of Babuska and coworkers, a novel self correcting error estimator strategy has been proposed and used in the simulation of flame propagation in a flowing mixture in Chapter 5. The grid generation algorithms and the predicted results are presented in Chapters 4 and 5.

CHAPTER 4

ADAPTIVE GRID TECHNIQUES FOR FLAME PROPAGATION PROBLEM
IN A STATIONARY MIXTURE.

4.1. Introduction :

This Chapter describes the h and h-p version adaptive mesh strategies and their application to the flame propagation and the test problems. *A posteriori* error criteria based on residual error have been used and tested numerically for the time dependent meshes. A statistical approach for the mesh refinement in h and h-p version has been explained in detail. The decisions for applying h-p refinement strategy and the conversion of higher degree to linear degree approximation in the h-p version have been explained. The effectiveness of the *a posteriori* error estimator has been studied by comparing the true and the computed errors at various times for the test problem in both the refinement methods. The conclusions drawn from the behaviour of various types of errors have been discussed. An attempt has been made to construct computationally inexpensive meshes for uniform front motion by the grid velocity concept. The performance of the h, h-p and grid speed schemes have been compared for both the problems.

4.2. History of error based adaption strategies :

The adaptive construction of meshes for the numerical solution of time-dependent partial differential equations has attracted wide interest in the recent past. Particularly appealing are the methods which involve the equi-distribution of some

measure of error, since they provide a well-graded mesh along with control over the solution error. Several schemes have been proposed based on the finite difference and the finite element approaches. In the following subsections, we shall briefly discuss some of the earlier work on adaptive mesh generation strategies.

4.2.1. Adaptive strategies by Finite Difference Methods :

For finite difference applications, several authors (Rai and Anderson, 1982; Brackbill, 1982; Dwyer and Sanders, 1983a; Gnoffo, 1982; Nakahashi and Deiwert, 1985; Saltzman and Brackbill, 1982; and Thompson et al., 1985b) have employed the structured grid generation techniques based on curvilinear co-ordinate lines, in their adaptive schemes. In these methods, the grid point locations are determined with the goal of minimizing some measure of error such as truncation error. The grid points are moved along the curvilinear co-ordinate lines till the error measure or the gradient is equi- distributed.

Some researchers (Anderson, 1983a,b; Rai and Anderson, 1981; Greenberg, 1985; Eiseman, 1983,1985) have proposed dynamically adaptive systems in which the grid points move in response to the development of the physical solution on the existing grid. Several problems in heat transfer, flame propagation and fluid mechanics have been considered in these works. The grid point distribution over the field readjusts dynamically to concentrate nodes in regions of larger solution variation as they develop, without reliance on prior knowledge of the location of such regions. The grid point distribution is also expected to retain a sufficient degree of smoothness and not lead to an excessively skewed grid,

else the truncation error will increase (Mastin, 1978; Thompson, 1982a, 1984, 1985d).

For the time dependent problems, some authors (Ghia et al., 1983; Dwyer et al., 1982; Dwyer, 1983b; Gnoffo, 1982) have adopted the strategy in which the the grid changes at each time step. In this approach the grid and the numerical solution have been solved separately. The time derivatives are transformed so that no interpolation onto the new grid is necessary. Some methods, however, change the grid at selected time steps and here interpolation must be used to transfer the values from the old grid to the new grid. Both the approaches have been discussed in detail by Thompson et al. (1982b). The variable used for grid adaption may differ from problem to problem; for instance, the grid could be adapted according to the velocity gradient in a flow problem and the temperature gradient in a flame propagation study. Equi- distribution concepts based on variational methods have been widely used in the recent past for grid adaption and these have been summarized in many survey papers and books (Hawken, 1987; Thompson et al., 1982b, 1984 ,1985a-c; Sengupta et al. 1988).

4.2.2. Adaptive strategies by the Finite Element Methods :

For finite element solution techniques, well developed mathematical theories are available for estimating the spatial discretization error in elliptic/ parabolic problems. For time-dependent problems, space-time finite element schemes are also available (Oden and Demkowicz, 1986 b, c; Babuska and Rheinboldt, 1982a; Davis and Flaherty, 1982; Gelinas et al, 1981). In the space- time approach, the mesh generation procedure depends upon

the minimization of the L_2 -error of the computed solution at a particular time. However, this procedure is computationally expensive as it involves the solution of non-linear equations.

Miller and Miller (1981 a, 1981b) have developed the moving finite element method in which the grid point locations are solved as additional variables by invoking the Galerkin formulation. In this approach, the residual is required to be orthogonal to all the basis functions of both the solution and the grid. Also, the relative motion as well as the minimum separation distance between grid points are controlled up to some extent.

Dupont (1982) has studied the finite element methods for which the underlying function spaces change with time. The error estimates and their convergence properties have been well-studied for the smooth solutions of parabolic problems in one space dimension and have been applied to fluid mechanics problems. Further, the types of continuous and discontinuous mesh modifications have been examined and certain symmetric a priori error estimates have been provided.

In the context of FE solutions, error estimates with certain desired properties have been developed for many classes of problems and these have been employed in the design of effective grid adaption procedures (Babuska and Rheinboldt, 1978a,b; 1982a, Babuska and Gui, 1986; Bank and Weiser, 1985; Gago et al., 1983). A *posteriori* error estimates based on the residuals have been proposed by Babuska and Rheinboldt (1978 a, b) for linear elliptic problems and have been widely used by many authors in finite element analysis. But for non-linear problems whose exact solution is not known, it can not always be guaranteed that the

error is bounded by the residual in some appropriate chosen norm, although such a feature is normally observed from the point of view of computations.

Several authors (Babuska and Dorr, 1981b; Babuska et al., 1981c, 1984; Demkowicz et al., 1989a; Gui and Babuska, 1986a,b,c; Oden et al., 1989; Szabo, 1986; Zienkiewicz et al., 1989) have worked on h and h - p refinement techniques for the solution of linear elliptic boundary value problems. These refinement techniques are commonly used in finite element modelling, for enhancing the convergence properties of the solution and reducing the error in the solution. Besides, these adaptive procedures allow not only the final global energy norm error to be well-estimated, but in addition, give a nearly optimal mesh. The desired accuracy can always be obtained with suitable h - p refinements and the rate of convergence of the adaptive h - p version analysis has also been analysed. It has been suggested that for high accuracy, the use of p version or h - p version adaptive analysis with a simple error estimator is most appropriate (Gui and Babuska, 1986 a, b, c; Zienkiewicz et al., 1989). Gui and Babuska (1986 a, b, c) have shown that in h - p refinement, it is possible to achieve an exponential rate of convergence to the exact solution for a large class of problems, even those including corner singularities. The p version usually shows a higher rate of convergence than the h version and its implementation in different adaptive situations are well documented. However, the practical implementation of an optimal h - p strategy is difficult and has not yet been achieved.

Bieterman and Babuska (1982a, b) have studied a variation of

the classical method of lines (MOL) in which the space- grid is updated in an adaptive procedure. In their adaptive FEMOL algorithm, grids change discontinuously in time as nodes are added or deleted. The primary goal of the adaptive FEMOL is to keep the space discretization error less than a user specified tolerance at all times. The *a posteriori* error estimator used here is similar to that proposed by Babuska and Rheinboldt (1978a), for linear PDE's. The most important part of the grid construction is a pattern recognition procedure which has been used to determine the "shape" of the grid. The process is made efficient, by controlling the error and monitoring the ODE time step-size information.

4.3. Problem Description :

In this chapter, the error estimators developed by Bieterman and Babuska (1982 a, b) have been used to simulate one dimensional front propagation problems. Both h and h-p methods have been employed and the equi-distribution of a *posteriori* error has been achieved through the statistical procedure proposed by (Carey and Oden, 1984a; Oden and Demkowicz, 1986a). Detailed comparisons are presented regarding the grid adaption capabilities of the h and h-p version strategies in terms of error behaviour, computational time/space requirements etc. The results from these two strategies are also compared with those of an inexpensive grid generation procedure based on the grid speed approach.

Two specific application problems have been considered here which display the feature of a sharp moving front in a stationary medium. The first is a linear test problem with a known source

equations (4.1a-c) is given as below.

$$u(x,t) = 1/2 + 1/2 \tanh [2\beta(x - Ct)] \quad (4.2)$$

This solution represents a wave front with approximate width $(2\beta^{-1})$ that moves in the positive x - direction with a speed C , as shown in Fig. 4.1. Apart from a thin moving front, it also includes decay and sharp transition regions in time and thus, has a form similar to those arising in flame or shock front propagation problems. Bieterman and Babuska (1982a,b) have implemented a variation of the classical method of lines (MOL) for the solution of the above type of problems in which the space grid is updated adaptively.

4.3.2. Description of the Flame propagation problem :

One- dimensional pre-mixed flame propagation in a stationary gaseous mixture is considered. The details of the formulation have already been explained in the Chapter 3 and here the governing equations are recalled, for the sake of convenience. The transient heat and species diffusion equations for the flame propagation are given in a normalized form as :

$$\left. \begin{aligned} \frac{\partial T}{\partial t} - \frac{\partial^2 T}{\partial x^2} &= f(\rho, T) \\ \frac{\partial \rho}{\partial t} - \frac{\partial^2 \rho}{\partial x^2} &= -f(\rho, T) \end{aligned} \right\} \quad t > 0, x \in (0,1) \quad (4.3)$$

with the same boundary and initial conditions as mentioned in equations (3.1b-c) of Chapter 3. The above equations can be expressed in a more general form as follows:

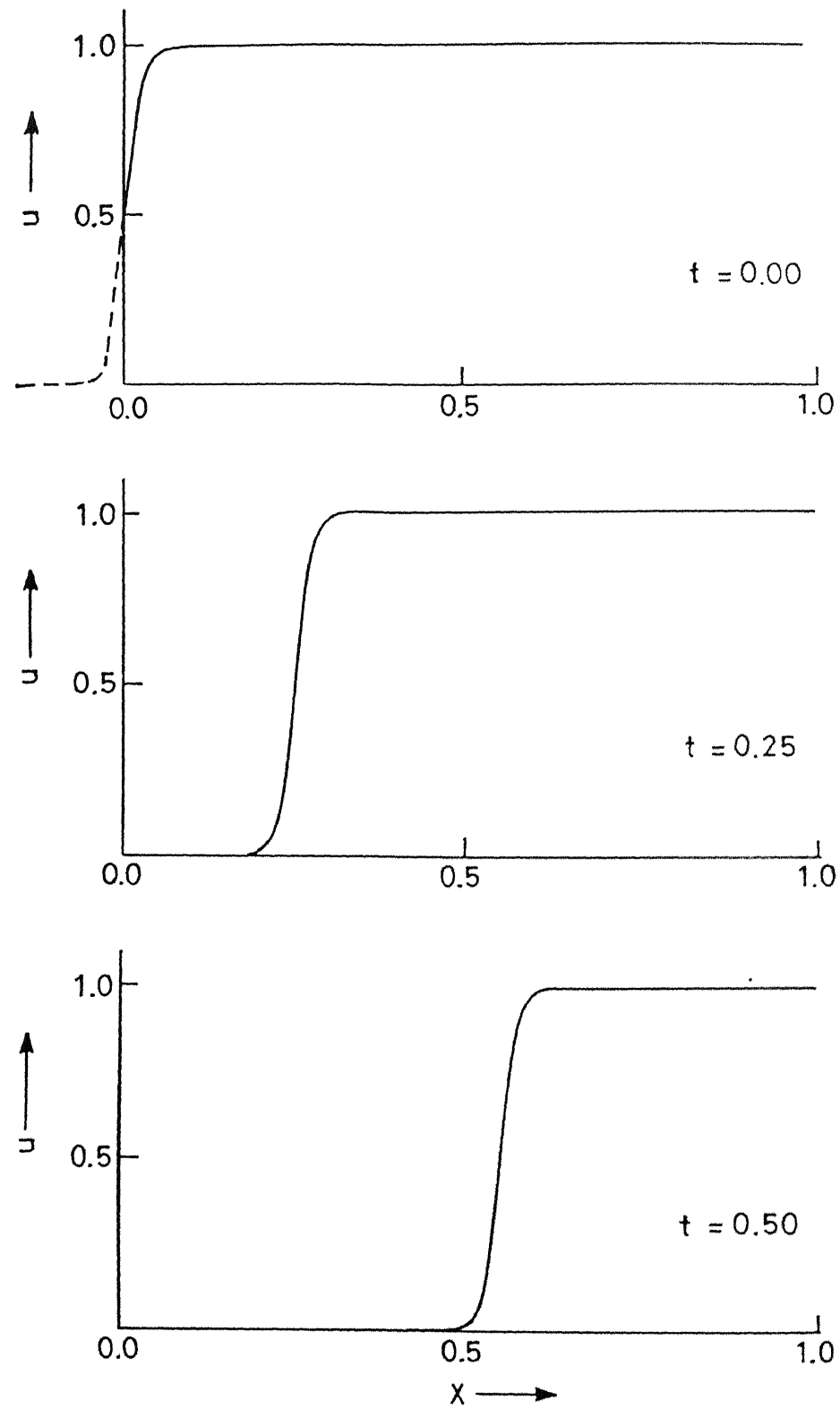


Fig. 4.1 TEST PROBLEM SOLUTION PROFILE AT DIFFERENT TIMES

$$(u_1)_t - \left(a_1(x) (u_1)_x \right)_x = f_1(x, t, u) \quad (x, t) \in \Omega \times (0, t_f],$$

(4.4)

$l = 1, \text{NPDE}$

For simplicity, we shall denote the Test problem (i.e. equation 4.2) as problem (T) and the flame propagation problem in the absence of flow (i.e. equation 4.4) as problem (F) in this chapter.

4.3.3. Adaptive grid notation :

Here, we briefly describe the notations employed to solve the evolution problem on a given time interval, say $[t_0, t_f]$, by the adaptive h and h - p refinement algorithms. For the sake of convenience, we consider the sub interval $[t_i, t_{i+1}]$ of the global time interval $[t_0, t_f]$ and present the notation used for all the necessary variables over this time step. Here, the subscript variable i in the quantity t_i takes values $0, 1, 2, \dots, f$ where t_0 is the starting time and t_f is the final time of the evolution problem.

Approximation Spaces and the Solution :

First, we recall the finite dimensional solution space $S_h^{r,k}(\Delta)$ which contains the approximation to $u_1(x, t)$ of the equation (4.4) for each l ($1 \leq l \leq \text{NPDE}$). It is evident from the earlier discussion that the linear degree approximation of finite element solution space, i.e.,

$$\left(S_h^{r,k}(\Delta); \quad r = 1, \text{ and the vector } k = (1, 1, \dots, 1) \right) \quad (4.5a)$$

is to be considered in the h refinement adaptive grid algorithm.

For simplicity, we shall denote this finite dimensional solution space i.e.,

$$S_h^{r,k}(\Delta) \text{ as } \mathcal{S}(\Delta). \quad (4.5b)$$

In the time interval $[t_1, t_{1+1}]$, we write

$$\mathcal{S}^{1,m}(\Delta) \quad (4.5c)$$

for the finite dimensional solution space at the m^{th} adaption at $t = t_1$, which contains the approximation to the solution

$$u_1(x,t) \text{ for } t \in [t_1, t_{1+1}] \text{ and for each } i \ (1 \leq i \leq NPDE).$$

Further, the dimension of $\mathcal{S}^{1,m}(\Delta)$ is $N_n^{1,m}$ which represents the number of basis functions and $N_e^{1,m}$ is the number of elements in the computational domain $\Delta^{1,m}$. The basis functions are denoted by

$$\psi_{1,j}^{1,m}(x), \quad (j = 1, \dots, N_n^{1,m}) \quad (4.5d)$$

$$\text{and } \mathcal{S}^{1,m}(\Delta) = \left\{ U_{1,j}^{1,m}(x,t) \mid U_{1,j}^{1,m}(x,t) = \sum_{j=1}^{N_n^{1,m}} U_{1,j}^{1,m}[t] \psi_{1,j}^{1,m}(x) \right\} \quad (4.5e)$$

where $U_{1,j}^{1,m}[t]$ are the nodal values of the approximate solution $U_{1,j}^{1,m}(x,t)$. Usually, with each approximation space $\mathcal{S}^{1,m}(\Delta)$ we associate a real vector

$$\bar{h}^{1,m} = \left(h_1^{1,m}, h_2^{1,m}, h_3^{1,m}, \dots, h_{N_e^{1,m}}^{1,m} \right) \quad (4.5f)$$

where each component $h_j^{1,m}$ ($1 \leq j \leq N_e^{1,m}$) represents the mesh size for the element $\Omega_j^{1,m} \left(= (x_j^{1,m}, x_{j+1}^{1,m}) \right)$ of the computational grid $\Delta^{1,m}$. Adaptive grid procedures typically employ a sequence of finite dimensional spaces

$$\varphi^{1,1}(\Delta), \varphi^{1,2}(\Delta), \dots, \varphi^{1,M}(\Delta),$$

with computational space meshes

$$\Delta^{1,1}, \Delta^{1,2}, \dots, \Delta^{1,M}.$$

Here, $\varphi^{1,M}$ is the space in which the final approximation is sought, with coarser spaces $\varphi^{1,1}, \varphi^{1,2}, \dots, \varphi^{1,M-1}$. In adaptive processes, however, there are no auxiliary spaces and the relationship between spaces is more involved. The nested approximation of spaces is not assumed because it poses severe restriction and is usually not met if there is node removal.

In the following discussion, we may refer to the vector $U_1^{1,m}$ by its component $U_{1,j}^{1,m}(x,t)$ and the vector

$$U_1^{1,m}[t] = \left\{ U_{1,j}^{1,m}[t] \right\}_{j=1}^{N_n^{1,m}} \quad (4.5g)$$

Further, we denote by

$$\bar{U}_1^{1,m}(x,t), \quad \hat{U}_1^{1,m}(x,t) \text{ and } \hat{U}_1^{1,m}(x,t)$$

the computed solution, the initial condition and the starting guess solution available at the m^{th} adaption at the time instant t_1 for each i ($1 \leq i \leq \text{NPDE}$).

We also use the following notation for the h-p refinement method. Let p_n be the n^{th} component of the vector \bar{p} defined as:

$$\bar{p} = \left\{ p_1, p_2, \dots, p_{N_e^{1,m}} \right\}. \quad (4.5h)$$

Here, p_n represents the degree of finite element approximation considered in the p version refinement of the n^{th} element. In the case of h refinement components of the vector \bar{p} ($= \bar{u}$) have the value as 1. In an analogous way, we define the sequence of finite dimensional solution spaces as

$$\varphi^{1,1}, \bar{p}(\Delta), \varphi^{1,2}, \bar{p}(\Delta), \dots, \varphi^{1,M}, \bar{p}(\Delta).$$

For the h-p refinement scheme the approximate solutions for the initial condition, starting guess solution and the computed solution at $t = t_1$ for m^{th} adaption are denoted by

$$\bar{\theta}_1^{1,m}, \bar{p}(x,t), \quad \hat{U}_1^{1,m}, \bar{p}(x,t) \quad \text{and} \quad \bar{U}_1^{1,m}, \bar{p}(x,t)$$

The following notations about the errors will be used quite often. $\varepsilon^{1,m}(t)$ denotes the *a posteriori* error which is obtained by summing up the local elemental error indicators $\eta_{1,n}^{1,m}(t)$ for the n^{th} element $\Omega_n^{1,m}$ of the computational mesh $\Lambda^{1,m}(t)$ at $t = t_1$ for each n ($1 \leq n \leq NPDE$). Further, $E_n^{1,m}(t)$ denotes the error indicator for the n^{th} element $\Omega_n^{1,m}$, ($1 \leq n \leq N_e^{1,m}$) and $\mu^{1,m}$ the mean and $\sigma^{1,m}$ the standard deviation of the set of local error indicators $E_n^{1,m}(t)$; ($1 \leq n \leq N_e^{1,m}$) of all PDE's at the instant $t = t_1$. The above symbols defined for the h and h-p version methods have been summarized in Table 4.1.

Interpolation between Approximation Spaces :

In the description of adaptive grid procedure, we shall need the interpolation operator

$$g_{1,m}^{1,m+1}(\cdot) \left(= g_{1,m}^{1,m+1} \right)$$

which is a linear transformation

from $\mathcal{S}^{1,m}$ to $\mathcal{S}^{1,m+1}$ such that $g_{1,m}^{1,m+1} \left(U^{1,m}(x,t) \right)$ is "close" to the approximation $U^{1,m+1}(x,t)$. In the nested or equal case $\mathcal{S}^{1,m} \subseteq \mathcal{S}^{1,m+1}$, it is natural to take $g_{1,m}^{1,m+1}$ as the identity operator \mathcal{I} (although this is not always the best choice). The interpolation operator $g_{1,m}^{1,m+1}$ is useful for the interpolation of solution variable from the m^{th} adaption to the $(m+1)^{th}$ adaption at each time $t = t_1$. In a similar fashion, we

| Variable | h version | | hp version | |
|------------------------------------|-----------------------------|-------------------------------|--|--|
| Time | t_i | | t_i | |
| Adaption | m | m + 1 | m | m + 1 |
| Computational mesh | $\Delta^{i,m}$ | $\Delta^{i,m+1}$ | $\Delta^{i,m}$ | $\Delta^{i,m+1}$ |
| Element 'n' | $\Omega_n^{i,m}$ | $\Omega_n^{i,m+1}$ | $\Omega_n^{i+1,m}$ | $\Omega_n^{i+1,m}$ |
| Finite dim. solun. space | $\mathcal{S}^{i,m}(\Delta)$ | $\mathcal{S}^{i,m+1}(\Delta)$ | $\mathcal{S}^{i,m}, \bar{\mathcal{P}}(\Delta)$ | $\mathcal{S}^{i,m+1}, \bar{\mathcal{P}}(\Delta)$ |
| Initial Condition | $\bar{U}_1^{i,m}(x,t)$ | $\bar{U}_1^{i,m+1}(x,t)$ | $\bar{U}_1^{i,m}, \bar{u}(x,t)$ | $\bar{U}_1^{i,m+1}, \bar{u}(x,t)$ |
| Semi discrete Approximation | $U_1^{i,m}(x,t)$ | $U_1^{i,m+1}(x,t)$ | $U_1^{i,m}, \bar{u}(x,t)$ | $U_1^{i,m+1}, \bar{p}(x,t)$ |
| Initial guess Solution | $\hat{U}_1^{i,m}(x,t)$ | $\hat{U}_1^{i,m+1}(x,t)$ | $\hat{U}_1^{i,m}, \bar{p}(x,t)$ | $\hat{U}_1^{i,m+1}, \bar{p}(x,t)$ |
| Computed Solution | $\bar{U}_1^{i,m}(x,t)$ | $\bar{U}_1^{i,m+1}(x,t)$ | $\bar{U}_1^{i,m}, \bar{p}(x,t)$ | $\bar{U}_1^{i,m+1}, \bar{p}(x,t)$ |
| Solution at Node 'j' | $U_{1,j}^{i,m}[t]$ | $U_{1,j}^{i,m+1}[t]$ | $U_{1,j}^{i,m}[t]$ | $U_{1,j}^{i,m+1}[t]$ |
| Linear basis fun. (j^{th} node) | $\psi_{1,j}^{i,m}(x)$ | $\psi_{1,j}^{i,m+1}(x)$ | $\psi_{1,j}^{i,m}(x)$ | $\psi_{1,j}^{i,m+1}(x)$ |
| Hierar.basis function | | | $\bar{\phi}_1^{i,m}(x)$ | $\bar{\phi}_1^{i,m+1}(x)$ |
| Unknowns for hierar basis | — | — | $\alpha_1^{i,m}(t)$ | $\alpha_1^{i,m+1}(t)$ |
| Dim of Linear basis fun. | $\mathcal{N}_n^{i,m}$ | $\mathcal{N}_n^{i,m+1}$ | $\mathcal{N}_n^{i,m}$ | $\mathcal{N}_n^{i,m+1}$ |
| Dim of hierar basis fun. | — | — | $\mathcal{N}_d^{i,m}$ | $\mathcal{N}_d^{i,m+1}$ |
| Elem(n) error for each PDE | $\eta_{1,n}^{i,m}$ | $\eta_{1,n}^{i,m+1}$ | $\eta_{1,n}^{i,m}$ | $\eta_{1,n}^{i,m+1}$ |
| Elem(n) error for all PDE's | $E_n^{i,m}$ | $E_n^{i,m+1}$ | $E_r^{i,m}$ | $E_r^{i,m+1}$ |
| Mean of the elem errors | $\bar{\epsilon}^{i,m}$ | $\bar{\epsilon}^{i,m+1}$ | $\bar{\epsilon}^{i,m}$ | $\bar{\epsilon}^{i,m+1}$ |
| Aposteriori Error | $\mathcal{E}^{i,m}$ | $\mathcal{E}^{i,m+1}$ | $\mathcal{E}^{i,m}$ | $\mathcal{E}^{i,m+1}$ |

Table 4.1. Schematic representation of variables used in the h and h-p version algorithm

also denote g_1^{i+1} as a linear interpolation operator from the time $t = t_1$ to t_{i+1} which will be useful for time marching in the h-p version. The order of interpolation may vary according to the circumstance. The global smoothness which depends upon the interpolation operator $g_{1,m}^{i,m+1}$ (or g_1^{i+1}) is not considered in the h refinement scheme. For example, in the finite element context, if polynomial interpolation is applied independently for each function $U_1^{i,m}(x,t)$, the smoothness is measured in terms of the smoothness of the individual functions which are defined on each element of the mesh.

4.3.4. Finite Element Formulation :

The details of the finite element formulation have been explained in the Chapter 3. Here, we state the necessary equations in the FE formulation which are useful in this chapter. The time Galerkin approximation of the solution of the system of equations (4.4) can be written as follows.

$$\begin{aligned} \langle \frac{\partial}{\partial t} U_1^i(.,t), V_1 \rangle + \langle a_1 \frac{\partial}{\partial x} U_1(x,t), \frac{\partial}{\partial x} V_1 \rangle = \langle \tilde{f}_1(x,t,U), V_1 \rangle \\ \text{for all } U_1, V_1 \in S_h^{r,k} ; t \in (0, t_f], \text{ and } i = 1, \dots, \text{NPDE.} \end{aligned} \quad (4.6a)$$

with suitably chosen initial conditions. The above system of equations (4.6a) can be written in a matrix form as

$$\begin{bmatrix} M_1 \end{bmatrix} \left\{ \frac{dU_1[t]}{dt} \right\} + \begin{bmatrix} K_1 \end{bmatrix} \{ U_1[t] \} = \{ F_1(U[t]) \} \quad \text{for } i=1, \dots, \text{NPDE} \quad (4.6b)$$

with the appropriate initial values. Employing the Crank-Nicolson time discretization scheme the matrix system for (4.6b) can be written as given below for each $t \in [t_{i-1}, t_i]$.

$$\begin{bmatrix} A_1^i \end{bmatrix} \left\{ U_1^i[t] \right\} = \left\{ G_1^i(U^i[t]) \right\} + \left\{ H_1^{i-1}(U^{i-1}[t]) \right\}; \quad (1 \leq i \leq NPDE) \quad (4.6c)$$

The details of the various quantities in the equations (4.5a,b,c) have been discussed in the Chapter 3.

4.4. Scope of the present study :

4.4.1. Objectives :

The basic aim is to develop a computational procedure which will meet the following goals :

- (G1). To obtain sufficiently accurate finite element solution to nonlinear partial differential equations representing flame propagation, where the accuracy is measured in some norm associated with the problem.
- (G2). To obtain optimal mesh distribution at each instant of time, which accurately captures the shape and location of the moving front.
- (G3). To develop h and h - p refinement strategies which meet certain tolerance criteria on the total *a posteriori* error for the whole solution domain.
- (G4). To develop a grid speed based adaptive strategy which is computationally inexpensive.
- (G5). To present a detailed study on the behaviour of error measures for the h and h - p refinement strategies and their application to Test problem and the Flame problem.
- (G6). To compare the relative performance of the h and h - p methods with each other and also with that of the grid speed strategy for the two problems.

depend on controlling the spatial discretization error through the predicted solution for each time. Also, the optimal mesh distribution is computed based on the principle of equi-distribution of spatial discretization error with the help of elemental error indicators. The statistical procedure proposed by Carey and Oden (1984a), Carey (1988) and Oden and Demkowicz (1991) has been employed for achieving equidistribution of spatial discretization error. Since, the exact spatial discretization error is not known for the flame problem, it is replaced by the *a posteriori* error estimate proposed by Bieterman and Babuska (1982a,b) which involves the finite element residue over each subinterval of the mesh. But, for the Test problem in which the exact solution is available, the performance of the *a posteriori* error has been evaluated by the various error measures described in Chapter 3.

4.4.2. Mathematical formulation of the objectives :

In this subsection, the objectives are expressed in the mathematical set up. Let us recall the *a-priori* error estimate which is given by

$$\|e(x,t)\| = \left\{ \int_{\Omega} a(x) \left(\frac{\partial}{\partial x} (e(x,t)) \right)^2 dx \right\}^{1/2} \quad \forall t \in (0, t_f) \quad (4.7)$$

The essential aim of the present work can be broadly classified into two parts.

- (a). Given a problem (T) with exact solution $u(x,t)$ and a norm $\| \cdot \|$, design an efficient adaptive algorithm for constructing a finite element mesh Δ such that

$$\|e(x,t)\| \leq \text{TOL}, \quad \forall t \in (0, t_f) \quad (4.8)$$

where $\bar{U}(x,t)$ is the finite element solution on the mesh Δ

and TOL is the given error tolerance.

The *a priori* error $\| e(x,t) \|$ is replaced by a *posteriori* error $\varepsilon(t)$ since the present adaptive algorithm is based on $\varepsilon(t)$.

(b). Given a problem (F) and a norm $\| \|$, design an efficient adaptive algorithm for constructing a finite element mesh Δ satisfying equation (4.8) for each i ($1 \leq i \leq \text{NPDE}$).

The *a priori* error $\| e(x,t) \|$ is substituted by a *posteriori* error $\varepsilon(t)$ since not only the present adaptive algorithm is based on $\varepsilon(t)$ but also the exact solution of the system (4.4) is not known.

The above aims (a) and (b) of the present work can be written in a simplified way which is as follows :

The main aim is to control the *a posteriori* error

$$\varepsilon(t) \leq \text{TOL}, \forall t \in (0, t_f) \quad (4.9)$$

and the perform mesh modification in order to achieve (4.9) at different time instants .

Description for each time :

The above aim has the following detailed description for each time interval; $[t_0, t_1]$; $[t_1, t_2]$; $[t_2, t_3]$; ..., $[t_{f-1}, t_f]$ of the total time interval $[t_0, t_f]$.

The adaptive algorithm seeks to construct a mesh

$$\Delta^{i,M} \text{ with mesh size vector } \bar{h}^{i,M}$$

and corresponding discrete FE solution

$$\bar{U}_1^{i,M,P}(x,t); \quad (1 \leq i \leq \text{NPDE})$$

such that

$$\varepsilon^{1,M}(t) \leq \text{TOL}, \text{ for each } t \in (t_1, t_{1+1}], \quad (4.10)$$

usually by constructing a sequence of meshes

$$\Delta^{1,1}, \Delta^{1,2}, \Delta^{1,3}, \dots, \Delta^{1,M} \quad (4.11a)$$

of mesh size vector

$$\bar{h}^{1,1}, \bar{h}^{1,2}, \bar{h}^{1,3}, \dots, \bar{h}^{1,M} \quad (4.11b)$$

with corresponding solutions

$$\begin{aligned} \bar{U}_1^{1,1}, \bar{P}(x,t), \bar{U}_1^{1,2}, \bar{P}(x,t), \dots, \bar{U}_1^{1,M}, \bar{P}(x,t); \\ \bar{U}_1^{1,m}, \bar{P}(x,t) \in \mathcal{S}^{1,m}, \bar{P}(\Delta). \end{aligned} \quad (4.11c)$$

The mesh $\Delta^{1,m+1}$ is constructed from

$$\bar{U}_1^{1,m}, \bar{P}(x,t) \in \mathcal{S}^{1,m}, \bar{P}(\Delta) \quad (4.11d)$$

by equidistribution of the contributions of different elemental error indicators

$$E_n^{1,m}(t); \quad (1 \leq n \leq N_e^{1,m})$$

to the *a posteriori* error

$$\varepsilon^{1,m}(t), \quad t \in [t_1, t_{1+1}].$$

Here, the solution $\bar{U}_1^{1,m}, \bar{P}(x,t)$ is obtained by solving the semi discrete problem

$$\begin{aligned} \left\langle \frac{\partial}{\partial t} U_1^{1,m}, \bar{P}(x,t), V_1^{1,m} \right\rangle + \left\langle a_1 \frac{\partial}{\partial x} U_1^{1,m}, \bar{P}(x,t), \frac{\partial}{\partial x} V_1^{1,m} \right\rangle \\ = \left\langle \tilde{f}_1(x,t, U_1^{1,m}, \bar{P}(x,t)), V_1^{1,m}, \bar{P}(x,t) \right\rangle \\ \forall U_1^{1,m}, \bar{P}(x,t), V_1^{1,m}, \bar{P}(x,t) \in \mathcal{S}^{1,m}, \bar{P}(t); \quad (x,t) \in \Omega \times (t_1, t_{1+1}] \\ (1 \leq i \leq \text{NPDE}), \quad \text{and } m = 1, \dots, M \end{aligned} \quad (4.11e)$$

with appropriate boundary and initial conditions. Here, all the components of the vector \bar{P} may take different integral values

depending on the refinement scheme. Clearly, our basic goal has two ingredients : Firstly, we want the adaptive algorithm to be reliable in the sense that the error control (4.9) is guaranteed by the construction. Secondly, we want the algorithm to be efficient in the sense that, the constructed mesh Δ is nowhere over refined. In order to achieve this, a priori estimate has been used in a certain sense directly for the problem (T).

4.5. h-version adaptive strategy :

In this section, the h - version grid refinement algorithm is presented. Various steps involved in the finite element solution and mesh modification are described in detail.

4.5.1. Description of the algorithm :

The algorithm for the time dependent evolution of the solution and the mesh distribution is described here. At the beginning of each time step, the initial data involves the specification of (i) initial condition (i.e., solution at previous time instant), (ii) initial mesh, and (iii) starting guess solution for the non-linear source term. After each adaption within a time step, the mesh distribution and the starting guess solution change, while the initial condition requires interpolation to the new mesh. Inner iterations are performed on each adapted mesh to ensure convergence of the non-linear finite element solver. On the basis of such finite element solutions a *posteriori* error is estimated with the help of local elemental error indicators and decisions for mesh modification are taken. If the total a *posteriori* error over the whole solution domain is less than a prescribed

tolerance, the grid adaption procedure is halted and the solution at this stage is accepted as the computed solution for the updated time level. The sequence of steps performed for each adaption cycle are enumerated below. These have also been shown in the form of a flow chart in Fig. 4.2.

Step 1 : Calculate the finite element solution at $t = t_1$:

Calculate the FE solution at $t = t_1$ for m^{th} adaption on the mesh $\Delta^{1,m}$ with given initial condition and the starting guess solution $t = t_1$ (initially with $m=1$ and $m=1$)

Step 2 : Calculate the a posteriori error estimate :

Calculate the a posteriori error estimate $\varepsilon^{1,m}$ at $t = t_1$ for m^{th} adaption with the computed solution $\bar{u}^{1,m}(x,t)$.

Step 3 : Check for the application of adaptive grid :

if $\varepsilon^{1,m}(t) \leq \text{TOL}$ then stop the adaption & go to step 5.
if $\varepsilon^{1,m}(t) > \text{TOL}$ then go to step (4) for adaptive grid.

Step 4: (a). Implementation of adaptive h-refinement scheme

- (i). For any $\Omega_n^{1,m} \in \Delta^{1,m}$, if $\text{MINTOL} < E_n^{1,m}(t) < \eta^{1,m}$ then RETAIN the element as it is.
- (ii). For any pair of elements, $(\Omega_n^{1,m}, \Omega_{n+1}^{1,m})$ if $E_n^{1,m}(t) \wedge E_{n+1}^{1,m}(t) \leq \text{MINTOL}$, then remove the common node between these elements.
- (iii). For any $\Omega_n^{1,m} \in \Delta^{1,m}$, if $E_n^{1,m}(t) > \eta^{1,m}$ then REFINES the element according to statistical procedure.

(b). Adaptive grid cycle :

Generate initial data on the new mesh $\Delta^{1,m+1}$

according to STEP 4a, increase the adaption counter

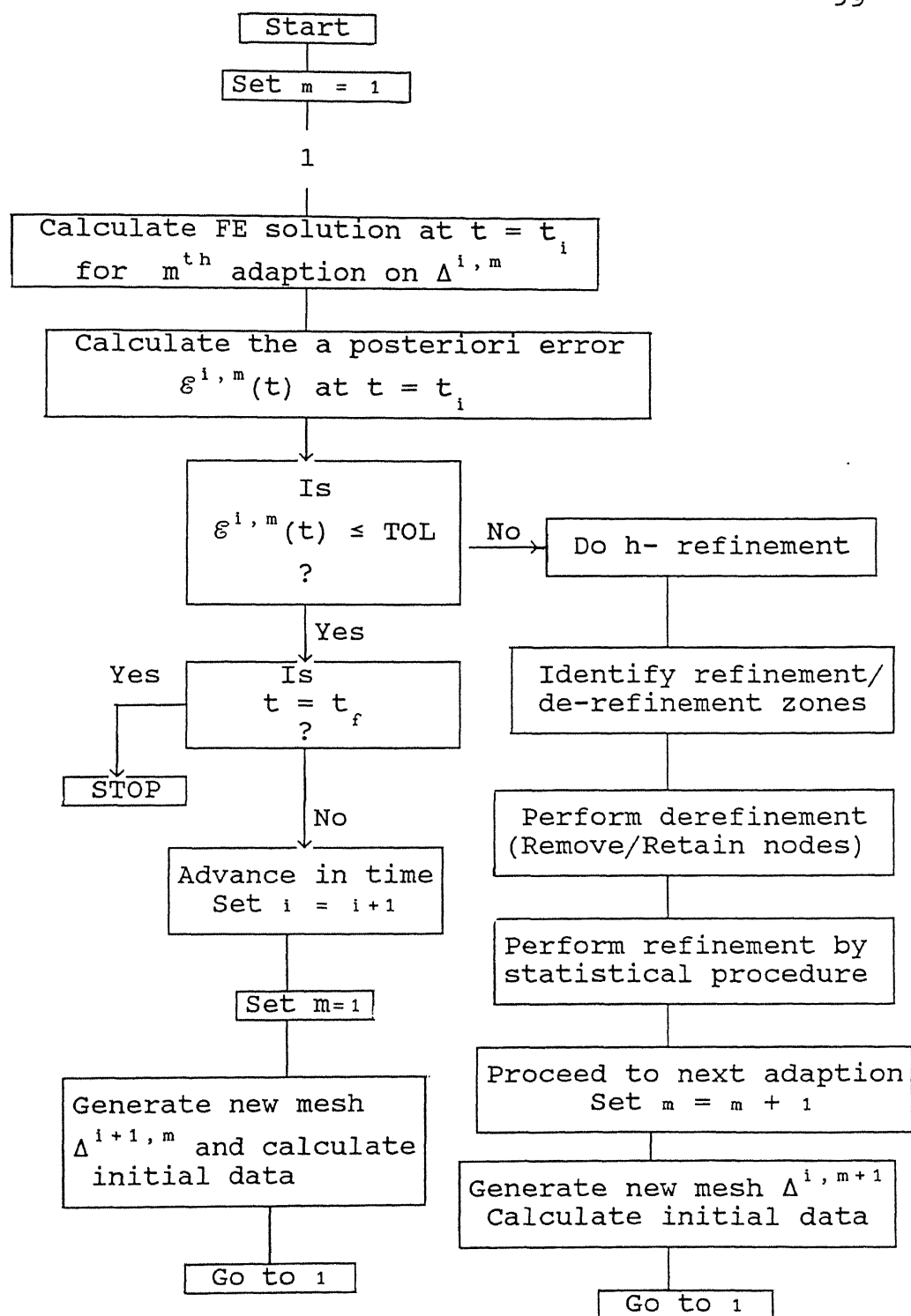


Fig. 4.2. Flow chart for h-version algorithm

(m) by one and go to step 1.

Step 5 : Time marching :

if $t < t_f$ then set $t_{i+1} = t_i + \Delta t$; construct and generate the initial data on the new mesh $\Delta^{i+1,1}$ at $t = t_{i+1}$, reset counter (m) to 1 and go to step 1.
if $t = t_f$ then stop the algorithm.

4.5.2. Important features of the algorithm :

Finite element solution :

The finite element solution procedure requires initial data such as initial condition and starting guess solution at each adaption of a time step. In the adaption cycle, the initial condition $\hat{U}_1^{1,m}(x,t)$ is obtained for the m^{th} adaption by interpolating the computed solution at the previous time level. For evaluating the non linear source term on the right hand side of the equations (4.5a) at $t = t_1$ an starting guess solution is needed. We choose the initial condition $\hat{U}_1^{1,m}(x,t) \in \mathcal{S}^{1,m}(\Delta)$ as the starting guess solution.

$$\left(\hat{U}_1^{1,m}(x,t) \in \mathcal{S}^{1,m}(\Delta) \right) \text{ at } t = t_1$$

for the initial mesh adaption. For the subsequent adaptations ($m > 1$) the freshly computed solution is used as the starting guess solution.

With the necessary initial data, the finite element solution for the matrix system of equations (4.5c) is obtained at each $t = t_1$ for each adaption according to the solution procedure described in section (3.3.3) of Chapter 3. Iterative schemes are required for solving the non-linear FE equations and execution of the adaptive mesh refinement, so that fully converged solutions

are not obtained in the FE solution procedure during intermediate stages of grid refinement.

A posteriori error calculation :

The *a posteriori* error estimate at $t = t_1$ for the m^{th} adaption of the system of equations (4.4) is given by

$$\varepsilon^{i,m}(t) = \left[\sum_{n=1}^{N_e^{i,m}} E_n^{i,m}(t) \right]^{1/2} = \left[\sum_{n=1}^{N_e^{i,m}} \left(\sum_{l=1}^{NPDE} |\eta_{l,n}^{i,m}(t)|^2 \right) \right]^{1/2} \quad (4.12a)$$

where, $E_n^{i,m}(t)$ is the local error indicator including all partial differential equations on the n^{th} element ($= \Omega_n^{i,m}$) and $\eta_{l,n}^{i,m}(t)$ is the local error indicator for l^{th} partial differential equation (4.4) alone. The local error indicator is given by the expression

$$|\eta_{l,n}^{i,m}(t)|^2 = \begin{cases} 0 & \text{if } \bar{a}_l^n = 0 \\ \frac{h^2}{12 \bar{a}_l^n} \int_{x_n^{i,m}}^{x_{n+1}^{i,m}} |r_{l,n}^{i,m}(x,t)|^2 dx & \text{if } \bar{a}_l^n \neq 0 \end{cases} \quad (1 \leq l \leq NPDE) ; (1 \leq n \leq N_e^{i,m}) \quad (4.12b)$$

Here, $\bar{a}_l^n = a_l \{ (x_n^{i,m} + x_{n+1}^{i,m})/2 \}$, ($1 \leq l \leq NPDE$) and the function $r_{l,n}^{i,m}(x,t)$ is the residual of l^{th} partial differential equation (4.4) on the n^{th} element (neglecting the discontinuities at the nodal points). The global residual of the l^{th} partial differential equation (4.4) at $t = t_1$ is given by

$$r_1^{1,m}(x,t) = \frac{\partial}{\partial t} \bar{U}_1^{1,m}(x,t) - \frac{\partial}{\partial x} \left\{ a_1(x) \frac{\partial}{\partial x} \left(\bar{U}_1^{1,m}(x,t) \right) \right\} - \bar{f}_1^{1,m}(x,t, \bar{U}_1^{1,m}(x,t))$$

(4.12c)

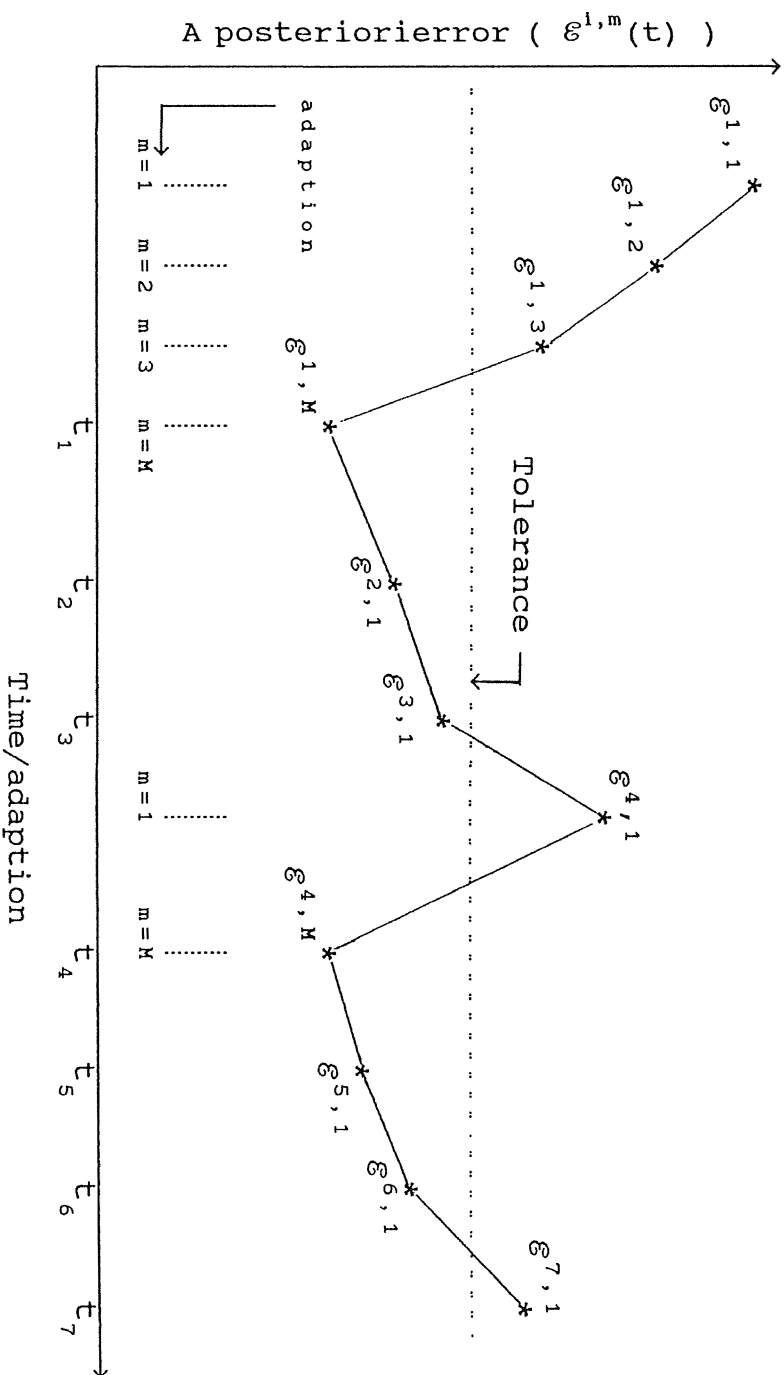
where, $\bar{U}_1^{1,m}(x,t)$ is the computed solution of i^{th} PDE at $t = t_1$ for m^{th} adaption. The local error indicator is evaluated via eight point Gaussian quadrature in eq. (4.12b) for each element.

Mesh refinement strategy :

Decisions concerning the mesh modification at each iteration depend upon the space discretization error. If the *a posteriori* error evaluated for a particular adaption at a particular time step does not satisfy the tolerance criterion, then we identify the subregions of the space mesh where the error in the predicted solution is large and refine such zones. Further, after the refinement, the necessary initial data on the refined mesh are calculated with the help of interpolation schemes. The iterative computational procedure for obtaining the solution is repeated on each new mesh until a specified stopping criterion is met. The typical variations of the *a posteriori* error with adaptations and time step are illustrated schematically in Fig. 4.3.

Statistical procedure and element divisions :

The particular scheme used here assesses and refines the mesh according to a statistical procedure (Carey and Oden, 1984a,b; Oden and Demkowicz, 1986c; Babuska and Rheinboldt, 1982a). One of the principle feature of this procedure is that a given element



with a large computed error, may be divided into several sub elements. The number of subdivisions is proportional to the magnitude of the local elemental error. The refinement and derefinement procedures are carried out on the basis of the local error exceeding or falling below the mean value of the local error indicators respectively. Elements with error less than the mean value are not refined. Besides, if the elemental error is a small fraction of the mean error for two adjacent elements, they are allowed to collapse into a single element by removing the common node. Such a statistical procedure involving simultaneous refinement and de-refinement at different parts of the solution domain is applied for each time step until the global a posteriori error meets the tolerance criterion. After a few iterations of this statistical distribution technique a well graded mesh results for each time step, while control on the total a posteriori error is provided by the proper selection of the tolerance limit TOL.

We shall now describe the details of the statistical procedure for the refinement of the mesh modification.

- (1). Consider the local elemental error indicators $E_n^{1,m}(t)$, $(1 \leq n \leq N_e^{1,m})$ evaluated at the m^{th} adaption of $t = t_1$, as the population to be ranked in the order of their magnitude. Taking the data set consisting of these error indicators as a random sample of given size drawn from a large population the mean and standard deviation are computed.
- (2). The standard deviation $\sigma^{1,m}$ is given by the relation

$$\sigma^{i,m} = \left(\sum_{j=1}^{N_e^{i,m}} (E_j^{i,m}(t) - \bar{\nu}^{i,m})^2 \right)^{1/2} / (N_e^{i,m} - 1) \quad (4.13)$$

(3). We classify the elements into various adaption sets based on the magnitude of the local error indicator $E_n^{i,m}(t)$ in relation to the mean $\bar{\nu}^{i,m}$ and the standard deviation $\sigma^{i,m}$. In the following sets, the variable 'n' varies from 1 to $N_e^{i,m}$.

$$\begin{aligned} \mathcal{P}_1 &= \left\{ \Omega_n^{i,m} : \text{MINTOL} < E_n^{i,m}(t) < \bar{\nu}^{i,m} \right\} \\ \mathcal{P}_2 &= \left\{ (\Omega_n^{i,m}, \Omega_{n+1}^{i,m}) : E_n^{i,m}(t) \wedge E_{n+1}^{i,m}(t) \leq \text{MINTOL} < \bar{\nu}^{i,m} \right\} \\ \mathcal{P}_3 &= \left\{ \Omega_n^{i,m} : h_n^{i,m} \leq \delta \right\} \\ \mathcal{P}_4 &= \left\{ \Omega_n^{i,m} : E_n^{i,m}(t) \geq \bar{\nu}^{i,m} \wedge h_n^{i,m} > \delta \right\} \\ \mathcal{P}_5 &= \left\{ \Omega_n^{i,m} : E_n^{i,m}(t) \geq \text{MINTOL} \wedge h_n^{i,m} > \delta \right\} \end{aligned} \quad (4.14)$$

where MINTOL is the prescribed tolerance for the removal of nodes in the domain $\Delta^{i,m}$ and δ is the prescribed minimum mesh size considered in the solution procedure. It is observed from the definition of these sets that all of them can never be empty.

(4). Define the "refinement intervals" on the distribution of elemental error indicators by :

$$\begin{aligned} I_1 &= [\bar{\nu}^{i,m}, \bar{\nu}^{i,m} + k\sigma^{i,m}], \quad I_2 = [\bar{\nu}^{i,m} + k\sigma^{i,m}, \bar{\nu}^{i,m} + 2k\sigma^{i,m}], \dots \\ , \dots, I_t &= [\bar{\nu}^{i,m} + (t-1)k\sigma^{i,m}, \bar{\nu}^{i,m} + tk\sigma^{i,m}], \dots \end{aligned} \quad (4.15)$$

where k represents the fraction of standard deviation defining each refinement interval.

- (5). The criteria for removing, retaining and refining the elements can be explained with the help of the sets defined above.

(a). REMOVE :

For any n , $1 \leq n \leq N_e^{1,m}$ if $(\Omega_n^{1,m}, \Omega_{n+1}^{1,m}) \in \mathcal{P}_2$ then the node which is common to both the elements is removed.

(b). RETAIN :

For any n , $1 \leq n \leq N_e^{1,m}$, if $\Omega_n^{1,m} \in \mathcal{P}_1 \cup \mathcal{P}'_1$, then the element $\Omega_n^{1,m}$ is retained as such, provided neither $(\Omega_{n-1}^{1,m}, \Omega_n^{1,m})$ nor $(\Omega_n^{1,m}, \Omega_{n+1}^{1,m}) \in \mathcal{P}_2$

(c). REFINE :

For any n , $1 \leq n \leq N_e^{1,m}$, if $\Omega_n^{1,m} \in \mathcal{P}_4$ then we calculate j for which $E_n^{1,m} \in I_j$ and the element is refined j times by the statistical procedure.

(d). MINIMUM GRID SPACE :

For any n , $1 \leq n \leq N_e^{1,m}$, if $\Omega_n^{1,m} \in \mathcal{P}_3$ then the element is not considered for refinement.

(6). Computational implementation :

Now, we implement all the steps (a), (b), (c) and (d) as described above for all the elements in the domain $\Delta^{1,m}$. First, the unnecessary nodes on the mesh $\Delta^{1,m}$ are systematically removed applying step (a). Some elements of mesh $\Delta^{1,m}$ are refined according to step (c) and the degree of refinement is based on the statistical procedure as explained in the step (4). In the computational experiments,

the degree of refinement (i.e., the number of divisions introduced) is selected on the basis of the magnitude of $j(n)$ for the n^{th} element which is defined as

$$j(n) = (E_n^{i,m} - r^{i,m}) / k \sigma^{i,m} \quad (4.16)$$

Besides, some elements on the mesh $\Delta^{i,m}$ have been retained according to step (b) and step (d). Note that the restrictions have been placed on the minimum spacing of grid points in order to avoid the collapse of the mesh. In the actual computations, the value of k has been taken as one and the value of the $j(n)$ has been restricted to small numbers (say 1 or 2) in the wave front region in order to avoid the dense refinement on the mesh $\Delta^{i,m}$.

In the statistical procedure, the mean values of the successive residuals or error distributions regress toward zero and simultaneously a natural gradation of the mesh is produced. The strategy gives equal weight-age to the error over small intervals with high gradients and to the error on a much larger element at low gradient in less sensitive regions. In practice, the scheme is not sensitive to the type of problem considered.

Construction of refined mesh ($\Delta^{i,m+1}$) data at $(m+1)^{\text{th}}$ adaption of $t = t_1$

In the beginning of the transient problem ($t \in (t_0, t_1]$) the initial condition on the initial mesh is known. This initial condition is chosen as the starting guess solution for solving the non-linear system of equations (4.5c) at $t = t_1$. Let, the initial condition and the starting guess solution be denoted by

$$\hat{U}_1^{1,m}(x,t) \in \mathcal{P}^{1,m}(\Delta) ; \quad \hat{U}_1^{1,m}(x,t) \in \mathcal{P}^{1,m}(\Lambda) \quad \text{at } t = t_1 - t_0 \quad (4.17a)$$

For the finer meshes ($m > 1$), the computed finite element solution for the m^{th} adaption $\bar{U}_1^{1,m}(x,t) \in \mathcal{P}^{1,m}(\Lambda)$ on the computational mesh $\Delta^{1,m}$ at $t = t_1$ is interpolated on the newly refined mesh points of $\Delta^{1,m+1}$ in the adaptive scheme to obtain the starting guess solution $\hat{U}_1^{1,m+1}(x,t) \in \mathcal{P}^{1,m+1}(\Lambda)$. In other words,

$$\hat{U}_1^{1,m+1}(x,t) = \mathcal{I}_{1,m}^{1,m+1} \bar{U}_1^{1,m}(x,t) \quad (4.17b)$$

where $\mathcal{I}_{1,m}^{1,m+1}$ is a linear interpolation operator from the m^{th} adaption to the $(m+1)^{\text{th}}$ adaption at the time instant t_1 . In a similar fashion, the initial condition on the finer meshes ($m > 1$) as can be obtained from

$$\hat{U}_1^{1,m+1}(x,t) = \mathcal{I}_{1,m}^{1,m+1} \{\hat{U}_1^{1,m}(x,t)\} \quad (4.17c)$$

respectively. In terms of adaption sets, on a particular element, the initial data can be retained as follows. Let an element $\Omega_n^{1,m+1}$ ($1 \leq n \leq N_e^{1,m+1}$) of the refined mesh $\Delta^{1,m+1}$. Then the following holds :

If $\tilde{\psi} \in \Omega_n^{1,m+1} \in \Delta^{1,m+1}$, then

$$\tilde{\psi} = \tilde{x} \in \Omega_{n'}^{1,m} \in \Omega^{1,m} \quad \text{for some } n' \quad (1 \leq n' \leq N_e^{1,m}).$$

Also, $\tilde{x} \in \Omega_{n'}^{1,m} \in \mathcal{P}_1$ or \mathcal{P}_2 or \mathcal{P}_3 or \mathcal{P}_4 or \mathcal{P}_5 .

In reality, $\tilde{x} \in \Omega_{n'}^{1,m}$ may or may not be a nodal point of $\Delta^{1,m}$.

Thus, the initial condition can be written as

$$\hat{\theta}_1^{i,m+1}(\tilde{y}, t) = \begin{cases} \theta_1^{i,m}(\tilde{x}, t) & \text{if } \tilde{x} \in \Omega_n^{i,m} \in \mathcal{P}_1 \text{ or } \mathcal{P}_2 \text{ or } \mathcal{P}_3 \text{ or } \mathcal{P}_5 \\ & (1 \leq i \leq \text{NPDE}) \\ g_{i,m}^{i,m+1}(\theta_1^{i,m}(\tilde{x}, t)) = \sum_{n'=1}^{N_e^{i,m}} \theta_{1,n'}^{i,m}[t] \psi_{1,n'}^{i,m}(\tilde{x}) \\ & \text{if } \tilde{x} \in \Omega_n^{i,m} \in \mathcal{P}_4 \text{ and } (1 \leq i \leq \text{NPDE}) \end{cases} \quad (4.17d)$$

In a similar fashion, one can write the starting guess solution as follows :

$$\hat{U}_1^{i,m+1}(\tilde{y}, t) = \begin{cases} \bar{U}_1^{i,m}(\tilde{x}, t) & \text{if } \tilde{x} \in \Omega_n^{i,m} \in \mathcal{P}_1 \text{ or } \mathcal{P}_2 \text{ or } \mathcal{P}_3 \text{ or } \mathcal{P}_5 \\ & (1 \leq i \leq \text{NPDE}) \\ g_{i,m}^{i,m+1}(\bar{U}_1^{i,m}(\tilde{x}, t)) = \sum_{n'=1}^{N_e^{i,m}} \bar{U}_{1,n'}^{i,m}[t] \psi_{1,n'}^{i,m}(\tilde{x}) \\ & \text{if } \tilde{x} \in \Omega_n^{i,m} \in \mathcal{P}_4 \text{ and } (1 \leq i \leq \text{NPDE}) \end{cases} \quad (4.17e)$$

Here, $g_{i,m}^{i,m+1}$ is the linear interpolation operator which is widely used in the finite element interpolation theory. Thus the calculations in the eqns. (4.17d-e) are repeated for all the nodal values of $\tilde{y} \in \Delta^{i,m+1}$ and the initial condition and starting guess solutions for $(m+1)^{\text{th}}$ adaption are obtained.

Time marching :

In the h- refinement technique, the approximation space considered is a finite dimensional space of piece-wise linear degree functions. Hence, in time marching, the information available at the preceding time step is fully utilized for the generation of necessary initial data. Thus the finally computed solution is given by

$$\bar{U}^{1,M}(x,t) \in \mathcal{P}^{1,M}(\Delta) \quad \text{at } t = t_1 \quad \text{on } \Delta^{1,M} \text{ of } [t_1, t_1] \quad (4.18a)$$

which is taken as the approximation for the initial condition

$$\bar{U}^{1+1,1}(x,t) \in \mathcal{P}^{1+1,1}(\Delta) \quad \text{at } t = t_{1+1} \quad \text{on } \Delta^{1+1,1} \text{ of } [t_1, t_{1+1}] \quad (4.18b)$$

Further, we choose

$$\Delta^{1,M} \equiv \Delta^{1+1,1} \quad \text{and} \quad \mathcal{P}^{1,M}(\Delta) \equiv \mathcal{P}^{1+1,1}(\Delta). \quad (4.18c)$$

Together with the above set of equations (4.18a-c) which consists of necessary data for the computation of the given evolution problem at the next time interval, the h refinement algorithm is fully defined.

As the present problem is nonlinear no analysis related to the choice of time step has been performed. However, very small time steps have been employed so that the accuracy of the solution is not affected by time discretization. In other words, we have assumed that the resulting transient ODE systems have been solved exactly.

4.5.3. Computational procedure

A computer code incorporating the h -version adaption algorithm has been written and validated for the test problem as well as flame propagation. A finite element formulation has also been developed to solve initial-boundary value problems for vector systems of partial differential equations in one space dimension and time. The solution algorithm employs Crank Nicolson method for time discretization with a piece-wise linear degree polynomial approximation for spatial interpolation. For the purpose of grid adaption and testing the solution convergence, we have assumed

that there is no time discretization error over the whole time span of the problem.

At each instant of time, the finite element discretization leads to the solution of a nonlinear system of algebraic equations. In the present algorithm, a simple local iterative (Picard) method has been used to solve the matrix system of semi discrete equations (4.4). It is to be noted that, the fully converged solution on the grid is not at all computed prior to refinement. However, as soon as inner iterations satisfy a simple form of convergence on the given mesh, the grid refinement is performed. The convergence of the iterative method did not pose any problem, as we have considered very small time steps with the tolerance value for convergence set equal to 0.0000001.

Following the standard finite element procedure, the element contributions for the matrix system of ordinary differential equations have been generated and used to calculate the corresponding global matrices. In the Crank-Nicolson time discretization, small time steps have been chosen in order to obtain accurate as well as good estimates of space discretization error. In practice, however, one wishes to take as large a time step as possible to cut down the computational expense. But, larger time steps, can introduce some unwanted numerical oscillations into the solution, in addition to decreasing the solution accuracy. For the test problem, we have chosen uniform time steps of 0.000001 throughout the time evolution process and for the flame propagation problems two different time steps namely, $t = 0.00001$ (before ignition) and $t = 0.000005$ (after ignition) have been used. The initial condition for each time

step, has been expressed by its piece-wise linear degree approximation. For time $t = 0$, the continuous initial data is converted into the approximate initial vector by restriction to the nodes x_j of the initial mesh. Besides, for each time instant, the solution on the preceding mesh is interpolated suitably to obtain an approximation to the initial condition on the newly introduced nodes of the refined mesh.

In the test problem as well as the flame problem the following tolerance parameters have been considered through out the problem. For the de-refinement step in the adaptive procedure, local tolerance parameter (REMOVE TOL) has been chosen as 0.0000001 for the removal of any node common to two adjacent elements, and the local tolerance parameter (RETAIN TOL) has been taken 0.000001 for retaining the elements . Besides, the global *a posteriori* error tolerance(GLOBAL TOL) has been set equal to 0.075, which is used for accepting the converged solution of a time step. In the refinement step of each adaption, a combination of one/three/seven new nodes have been introduced in various elements according to the statistical procedure for adaptive refinement during the ignition phase or start-up transient propagation phase. But, in the wave front / flame front region, the refinement is confined to one or three new nodes in any element. This restriction is considered in order to avoid over refinement.

The error estimators and the nonlinear source term in the flame propagation problem have been computed by eight point numerical quadrature. The decisions concerning the mesh modification are made adaptively and automatically in the

programme at various stages of the operations. All computations have been carried out in double precision arithmetic in order to avoid the round off errors.

4.6. hp-version adaptive strategy :

4.6.1. Description of the algorithm :

In the h-p method, the p -version and the h -version approaches are used alternatively. In a routine way the FE solution is computed at a given time and the *a posteriori* error is calculated. If the tolerance criterion on a *a posteriori* error estimate is not satisfied at time $t = t_i$ for the m^{th} adaption, then either p refinement or h-p refinement is performed. The algorithm identifies regions where the errors are locally large using statistical procedure. The problem is first solved with linear degree finite element approximation at $t = t_i$ on the initial mesh. The de-refinement procedure is applied during the initial adaptations for various time intervals. The degree of finite element approximation is increased locally in the regions of large error and the computation is carried out till the *a posteriori* error tolerance criterion is met. But in case the error remains above the tolerance at $t = t_i$ and at the same time the prescribed maximum degree of polynomial approximation has been already reached in some elements, then the mesh size h is reduced locally in the regions where the higher order FE approximation is still needed. It is done by introducing new grid points in these elements along with simultaneous replacement of higher order approximation by piece-wise linear degree approximation.

After the construction of the refined mesh the problem is

solved with the linear degree approximation of solution on the new mesh. The *a posteriori* error estimate at $t = t_1$ on the refined mesh is also calculated. If the *a posteriori* error criterion is not still satisfied, then the degree of finite element approximation is increased till the error is brought down the tolerance level. In this way p or h-p strategy is applied judiciously to obtain the adaptive mesh as well as the FE solution.

As we advance in time from t_1 to t_{1+1} the approximate solutions for the initial condition and the starting guess solution are both chosen from the finite dimensional solution space of linear degree approximation in each element over the initial mesh at $t = t_{1+1}$. If the converged finite element solution on the mesh $\Delta^{1,M}$ at $t = t_1$ (which satisfies the *a posteriori* error criterion), contains higher degree approximation in some regions, then these regions are locally refined and the corresponding FE solutions are converted into equivalent linear degree approximation. This procedure is carried out in a way, as it has been explained for the h-p refinement. The interpolated computed solution on the refined mesh or the actual computed solution on the $\Delta^{1,M}$ at $t = t_1$ is chosen as the initial condition at $t = t_{1+1}$ the initial mesh $(\Delta^{1+1,1})$. In the following, various steps involved in the FE solution and mesh modification are described in detail. The flow chart for the h-p algorithm is shown in Fig. 4.4.

Step 1 : Calculate the finite element solution at $t = t_1$:

Calculate the FE solution at $t = t_1$ for m^{th} adaption on the mesh $\Delta^{1,m}$ with given initial condition and the

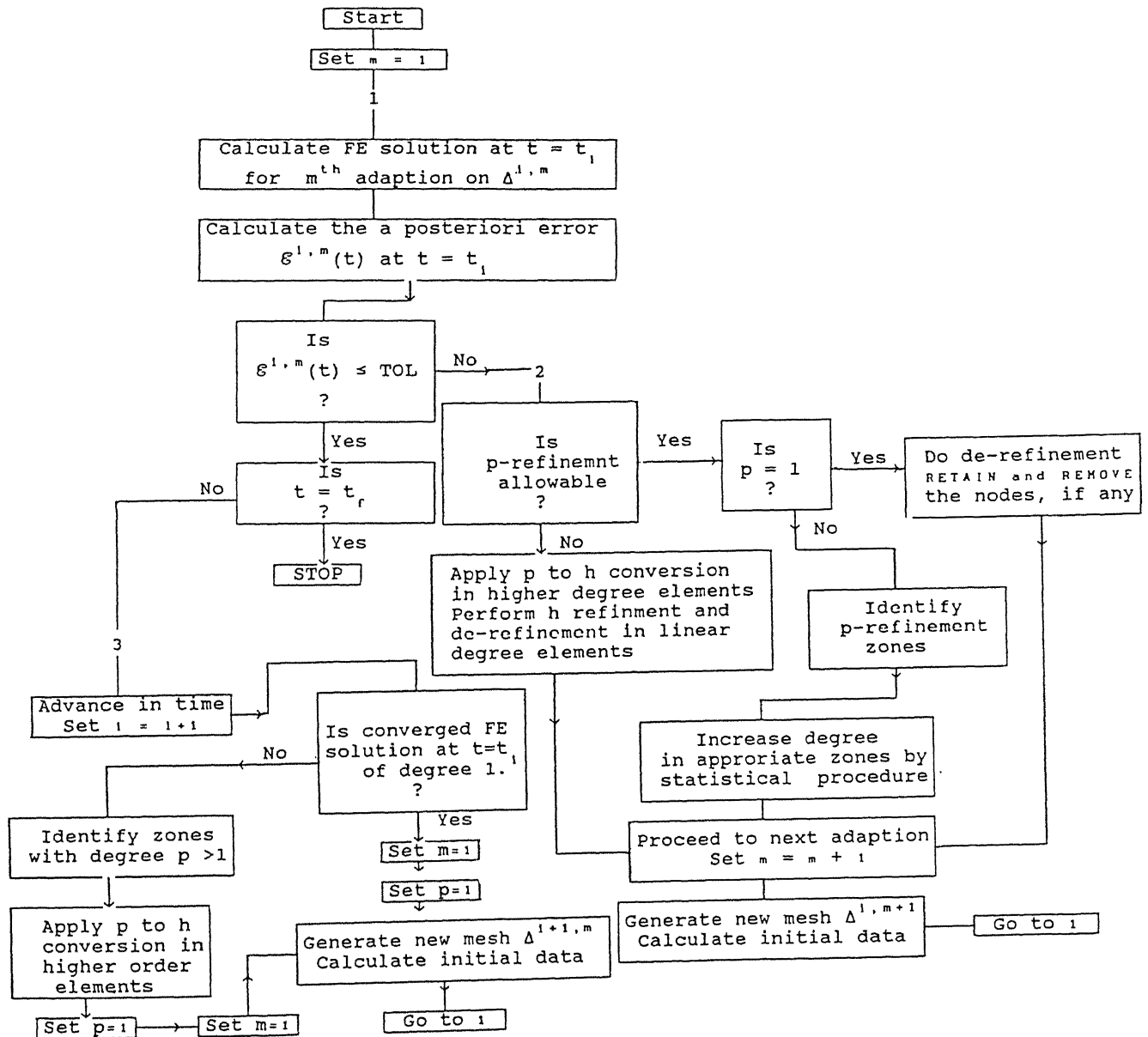


Fig. 4.4. Flow chart for h-p version algorithm

starting guess solution (initially $l = 1$ and $m = 1$).

Step 2 : Calculate the a posteriori error estimate :

Calculate the a posteriori error estimate $\mathcal{E}^{1,m}(t)$ with the computed solution $\bar{U}^{1,m}(x,t)$.

Step 3 : Check for the application of adaptive grid :

if $\mathcal{E}^{1,m}(t) \leq \text{Tol}$ then go to step 6 for advancing in time
 if $\mathcal{E}^{1,m}(t) > \text{Tol}$ then go to step 4 for the application of h-p refinement.

Step 4 : Implementation of hp refinement scheme :

For each element $\Omega_n^{1,m}$ $1 \leq n \leq N_n^{1,m}$ and do the following operations.

- (a). For any $\Omega_n^{1,m} \in \Delta^{1,m}$, if $E_n^{1,m}(t) < \tau^{1,m}$ then go to step 5a for de-refinement procedure.
- (b). For any $\Omega_n^{1,m} \in \Delta^{1,m}$,
 if $E_n^{1,m}(t) > \tau^{1,m}$ and $\text{Deg}(\Omega_n^{1,m}) < \text{MAXDEG}$ then go to step 5b for the application of p refinement.
- (c). For any $\Omega_n^{1,m} \in \Delta^{1,m}$,
 if $E_n^{1,m}(t) > \tau^{1,m}$ and $\text{Deg}(\Omega_n^{1,m}) = \text{MAXDEG}$ then go to step 5c for conversion of p to h refinement

Step 5 :(a). Application of de-refinement scheme.

The two operations RETAIN and REMOVE of nodes is done according to step(4) of h-refinement algorithm. After derefinement, go to step 6c.

(b). Application of p-version :

For for any $\Omega_n^{1,m}(t) \in \Delta^{1,m}$, if $\text{Deg}(\Omega_n^{1,m}) < \text{MAXDEG}$ then increase $\text{Deg}(\Omega_n^{1,m})$ by 1 and go to step 6a

(c). p version to h version conversion :

(i). If $\text{Deg}(\Omega_n^{1,m}) \geq 1$ and $E_n^{1,m}(t) < \tau^{1,m}$ then

retain the element configuration and consider the piece-wise linear degree approximation in $\Omega_n^{i,m}$

(ii). If $\text{Deg}(\Omega_n^{i,m}) = 1$ and $E_n^{i,m}(t) \geq \tau^{i,m}$ then apply h-refinement by the statistical procedure.

(iii). If $\text{Deg}(\Omega_n^{i,m}) \geq \text{MAXDEG}$ and $E_n^{i,m}(t) \geq \tau^{i,m}$ then refine the element into piece-wise linear elements by the p to h version conversion, according to step 6b.

Step 6 : Construct the refined mesh $\Delta^{i,m}$ data :

(a). p version :

Increase the adaption counter (m) by 1.

Incorporate the additional degrees of freedom and construct the initial data on $\Delta^{i,m}$. Then go to step 1.

(b). p version to h version :

Increase the adaption counter (m) by 1.

generate refined mesh $\Delta^{i,m+1}$ with piece wise linear FE approximation by p to h conversion. and go to step 1 for next adaption.

(c). De-refinement :

Increase the adaption counter (m) by 1.

construct the initial data on $\Delta^{i,m}$. Then go to step 1.

Step 7 : Time marching :

(a). If $t_1 < t_f$,

For each $\Omega_n^{i,m}(t) \in \Delta^{i,m}$, if $\text{Deg}(\Omega_n^{i,m}) > 1$

then refine $\Omega_n^{i,m}$ by p to h conversion and obtain

the piece-wise linear degree approximation in the refined elements of $\Omega_n^{1,m}$. Generate the initial data on the refined mesh $\Lambda^{1+1,1}$ and go to step 1.

(b) If $t_1 = t_f$ then stop the algorithm.

4.6.2. Important features of the algorithm :

Hierarchical forms of higher order elements :

The hierarchical basis functions considered in the present study, simply represent additive refinements of higher order over the linear degree FE approximation. Let us consider the generation of these basis functions for a general element as illustrated in Fig. 4.5.

The approximation over this element cannot be improved upon as the identification of the connecting nodes between elements guarantee C^0 continuity. However, we can achieve the hierarchical format over this expansion by preserving the required C^0 continuity of the linear approximation between elements. This condition can be satisfied by choosing basis functions which vanish at the end points of the interval. A particularly attractive class of polynomial functions which preserve C^0 continuity at the end points are related to the integrals of Legendre polynomials $P_p(\zeta)$ over the range $-1 \leq \zeta \leq 1$. Here we define the Legendre polynomial of degree p by

$$P_p(\zeta) = \frac{1}{p!} \frac{1}{2^p} \frac{d^p}{d\zeta^p} \left[(\zeta^2 - 1)^p \right] \quad (4.19a)$$

and integrating $2p P_p(\zeta)$ for $p = 1, 2, 3, 4$, from -1 to ζ , in turn gives $\tilde{\phi}_3, \tilde{\phi}_4, \tilde{\phi}_5$, and $\tilde{\phi}_6$ etc., which are used as basis functions of quadratic, cubic, fourth and fifth order

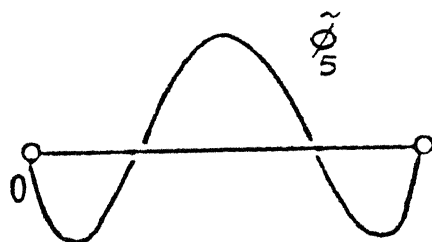
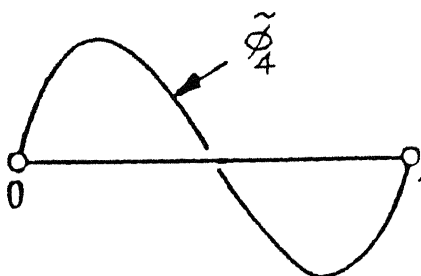
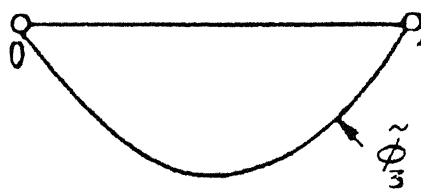
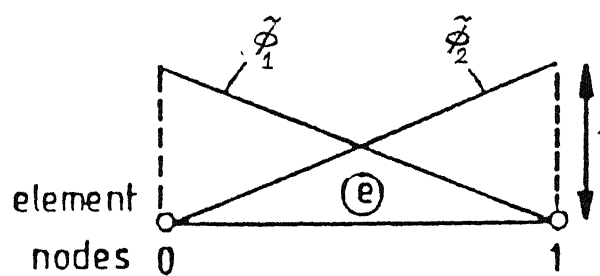


Fig. 4.5. Hierarchical element shape functions of nearly orthogonal form

approximation respectively. The structure of the elemental matrices arising from these hierarchical basis functions is simple, due to the orthogonality property of the Legendre functions over the interval $-1 \leq \zeta \leq 1$.

By means of such hierarchic nesting of the basis functions, one is able to utilize the element calculations of previous adaptations repeatedly up to some extent. The key feature to be observed here is that the addition of higher order terms does not alter the entries corresponding to the lower order basis functions. This implies that a large block sub matrix from the previous level remains unchanged. Using such hierarchical families of C^0 elements in one space dimension, the approximation for the unknown solution variable $u(\zeta)$ on a typical element $\hat{\Omega}$ for any partial differential equation can be written as follows :

$$u(\zeta) = u_1 \tilde{\phi}_1(\zeta) + u_2 \tilde{\phi}_2(\zeta) + \sum_{j=3}^p u_3^{(j-1)} \tilde{\phi}_j(\zeta) \quad (4.19b)$$

$$\zeta \in \hat{\Omega} = [-1, 1]$$

where $\tilde{\phi}_1$ and $\tilde{\phi}_2$ are the linear basis functions and $\tilde{\phi}_j$; $j > 2$ are correspond to higher degree Legendre polynomials and u_1, u_2 , are values of the unknown variable corresponding to the nodes 1 and 2 respectively. The additional degrees of freedom u_3^{j-1} ($1 \leq j \leq p$) enter as tangential derivatives at the mid side nodes, the nodal co-ordinate data on the end points of all elements is remain.

Finite element solution :

In the adaptive grid cycle, with the given initial condition

$$u_1^{1,m,u}(x,t) \in \mathcal{S}^{1,m,u}(\Delta) \text{ at } t = t_1.$$

and the starting guess solution

$$\hat{U}_1^{i,m,p}(x,t) \in \mathcal{S}^{i,m,p}(\Delta) \quad \text{at } t = t_i$$

on the mesh $\Delta^{i,m}$, the resulting non-linear system (4.4) is solved by the iterative scheme as described in Chapter 3. Here, the additional unknown variables introduced in the p-version have also been invoked in the FE formulation and solved together with the unknown variables $U_{1,j}^{i,m}[t]$ ($1 \leq j \leq N_n^{i,m}$). These additional unknown variables which are present in the approximation of the starting guess solution are initially chosen to be zero (the vector $p = u$). But for subsequent p version adaptations, these parameters need not be zero.

The a posteriori error evaluation :

In the h-p refinement method, p refinement and p-h refinement are employed during different adaptations of a particular time step. During p refinement, the higher degree approximations have been considered for the evaluation of local error indicators $\eta_{1,n}^{i,m}(t)$. Thus the a posteriori error estimate $\varepsilon^{i,m}(t)$ is given by the relation (4.12a) in which the error indicator $\eta_{1,n}^{i,m}(t)$ for the n^{th} element is given by

$$|\eta_{1,n}^{i,m}(t)|^2 \equiv \begin{cases} 0 & \text{if } \bar{a}_1^n = 0 \\ \frac{h^2}{12 p_n \bar{a}_1^n} \int_{X_n^{i,m}}^{X_{n+1}^{i,m}} |r_{1,n}^{i,m}(x,t)|^2 dx & \text{if } \bar{a}_1^n \neq 0 \end{cases} \quad (1 \leq n \leq NPDE)$$

(4.20a)

where p_n is the degree of approximation used in the n^{th} element, and \bar{a}_1^n is defined as earlier. Further, the residue in the

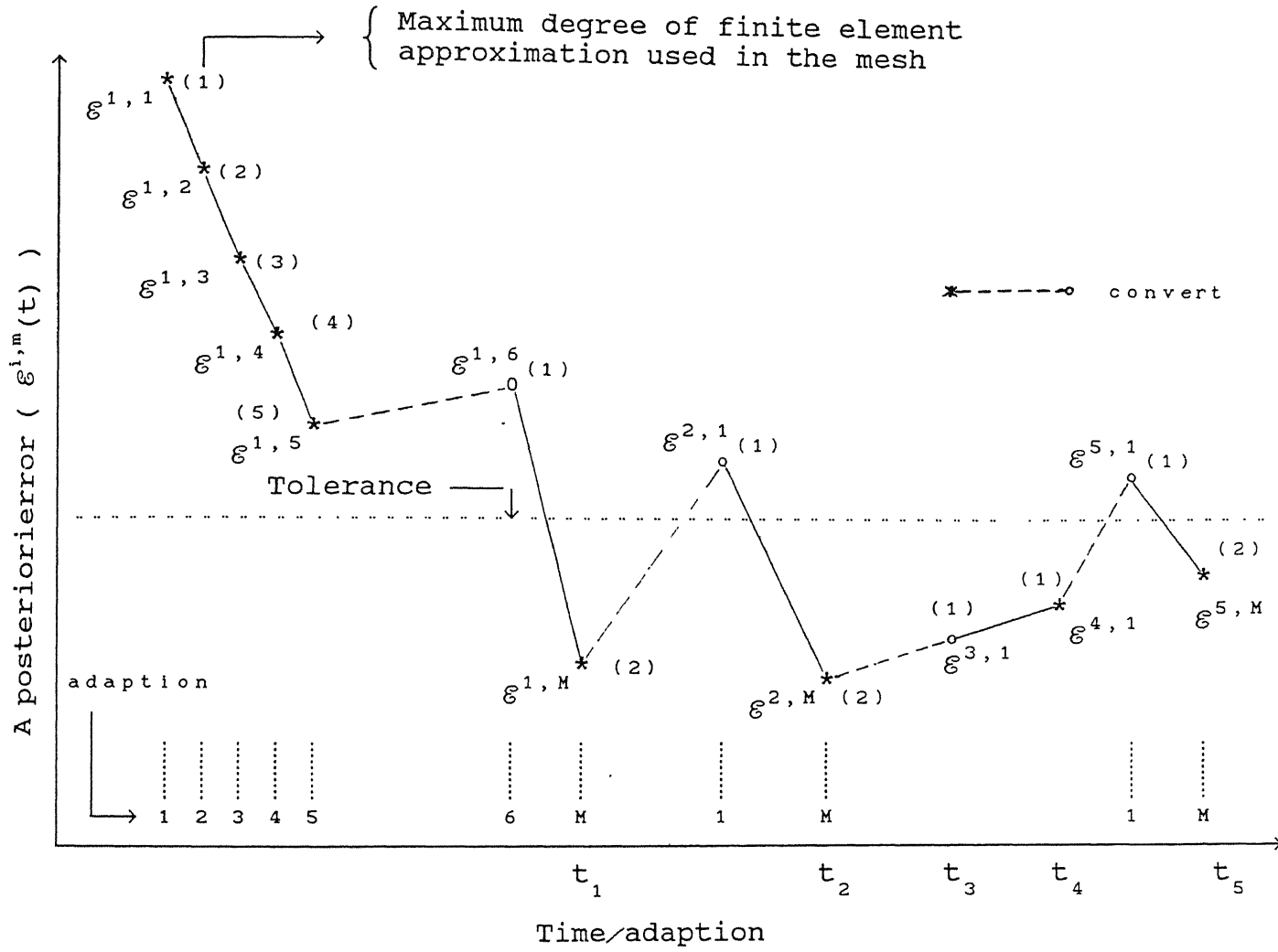


Fig. 4.6a. Schematic representation of the evolution of a posteriori error with adaptations/time in h-p version algorithm

based on the mean and the standard deviation of local error indicators. We define the following adaption sets. In all the following sets the variable 'n' varies from 1 to $N^{1,m}$

$$\begin{aligned}
 \mathcal{P}_1 &= \left\{ (\Omega_n^{1,m}, \Omega_{n+1}^{1,m}) : E_n^{1,m}(t) \wedge E_{n+1}^{1,m}(t) \leq \text{MINTOL} < \tau^{1,m}, \right. \\
 &\quad \left. \text{Deg}(\Omega_n^{1,m}) \text{ and } \text{Deg}(\Omega_{n+1}^{1,m}) = 1 \right\} \\
 \mathcal{P}_2 &= \left\{ \Omega_n^{1,m} : E_n^{1,m}(t) < \tau^{1,m} \text{ or } h_n^{1,m} < \delta \right\} \\
 \mathcal{P}_3 &= \left\{ \Omega_n^{1,m} : E_n^{1,m}(t) \geq \tau^{1,m} \wedge h_n^{1,m} > \delta, \text{Deg}(\Omega_n^{1,m}) = 1 \right\} \\
 \mathcal{P}_4 &= \left\{ \Omega_n^{1,m} : E_n^{1,m}(t) \geq \tau^{1,m} \wedge h_n^{1,m} > \delta, 1 < \text{Deg}(\Omega_n^{1,m}) < \text{MAXDEG} \right\} \\
 \mathcal{P}_5 &= \left\{ \Omega_n^{1,m} : E_n^{1,m}(t) \geq \tau^{1,m} \wedge h_n^{1,m} > \delta, \text{Deg}(\Omega_n^{1,m}) = \text{MAXDEG} \right\}
 \end{aligned} \tag{4.21}$$

- (2). For all elements, determine in which refinement interval the elemental error resides and refine the element accordingly either by p version or h version.

In the computational experiments, the de-refinement has been done only for elements on which the linear degree of FE approximation has been considered. It was discussed earlier that the piece-wise linear degree FE approximation has been used in the present algorithm on the initial mesh as well as on the mesh obtained after the conversion of p to h version. On such meshes with linear degree approximation, the derefinement procedure of the h version algorithm is applied first as explained below.

De-refinement :(i). REMOVE :

For any n , $1 \leq n \leq N_e^{i,m}$ if $(\Omega_n^{i,m}, \Omega_{n+1}^{i,m}) \in \mathcal{P}_1$ then the node which is common to both the elements is removed.

(ii). RETAIN :

For any n , $1 \leq n \leq N_e^{i,m}$, if $\Omega_n^{i,m} \in \mathcal{P}_2$ then the element $\Omega_n^{i,m}$ is retained as such provided neither $(\Omega_{n-1}^{i,m}, \Omega_n^{i,m})$ nor $(\Omega_n^{i,m}, \Omega_{n+1}^{i,m}) \in \mathcal{P}_1$.

p version refinement

For an element $\Omega_n^{i,m} \in \mathcal{P}_4$ on the mesh $\Delta^{i,m}$ and then increase $\text{Deg}(\Omega_n^{i,m})$ by one and recompute the FE solution on the refined mesh $\Delta^{i,m+1}$.

Although, the refined mesh $\Delta^{i,m+1} \equiv \Delta^{i,m}$, but the degree of FE approximation may be different between the elements of $\Delta^{i,m}$ and the element of $\Delta^{i,m+1}$.

hp version refinement :

Here, we describe the construction of refined mesh $\Delta^{i,m+1}$ from $\Delta^{i,m}$ for piece-wise linear degree FE approximation of solution. Since some elements of the mesh $\Delta^{i,m}$ are either linear or higher degree, we systematically apply refinement, and conversion of p to h version as discussed below.

(i). For $\Omega_n^{i,m} \in \mathcal{P}_3$ we find the integer j such that $E_n^{i,m}(t) \in I_j$ by the statistical procedure as explained in h version algorithm and subdivide the element in proportion with j

(ii). If $\Omega_n^{i,m} \in \mathcal{P}_2$ then consider linear degree FE

approximation and neglect the higher degree approximation in $\Omega_n^{1,m}$

(iii). If $\Omega_n^{1,m} \in \mathcal{P}_5$ then n^* grid points are introduced in $\Omega_n^{1,m}$ in such a way that the higher degree polynomial in $\Omega_n^{1,m}$ is approximated uniformly by piece-wise linear polynomials on the newly formed sub elements of $\Omega_n^{1,m}$

Now, we implement all the steps as described above for all the elements in the domain $\Delta^{1,m}$.

conversion of p version to h version :

The procedure for obtaining linear degree FE approximation from the higher degree p version over an element $\Omega_n^{1,m}$ and the conversion from p version to h version is described below. Since the construction of new mesh is based on the local gradient of the computed higher degree solution, a non-uniform refinement of each higher order element is obtained after conversion. Here, only one substantive variable τ is used. The substantive variable corresponds to the the temperature variable in the flame propagation problem. Let $\bar{U}_1^{1,m,\bar{p}}(x,t)$ denote the computed solution of temperature. Then

$\bar{U}_1^{1,m,\bar{p}}(x,t) \in \mathcal{P}^{1,m,\bar{p}}$ may be the approximate p-version FE solution in $\Omega_n^{1,m}$ of $\Delta^{1,m}$. Then on $\Delta^{1,m}$ we have

$$\bar{U}_1^{1,m,\bar{p}}(x,t) = \sum_{j=1}^{N_n^{1,m}} \bar{U}_{1,j}^{1,m} [t] \psi_{1,j}^{1,m}(x) + \sum_{j'=1}^{N_h^{1,m}} \alpha_{1,j'}^{1,m} \bar{\phi}_{1,j'}(x)$$

$$x \in \Delta^{1,m} \text{ and } 1 \leq i \leq \text{NPDE} \quad (4.22)$$

In the above equation (4.22), the unknown parameters $\alpha_{1,j}^{1,m}[t]$ ($j=1,2,\dots,N_h^{1,m}$) enter as additional degrees of freedom due to hierarchical p version approximation. The hierarchical basis functions $\bar{\phi}_{1,j}(x)$, $j = 1, \dots, N_h^{1,m}$, in this equation may be equal to quadratic/cubic/fourth/fifth order polynomials as described in the section (4.6.2). Further, note that the $\psi_{1,j}^{1,m}(x)$ ($1 \leq j \leq N_n^{1,m}$) are linear basis functions and $\bar{U}_{1,j}^{1,m}[t]$ ($1 \leq j \leq N_n^{1,m}$) are the corresponding solution values at the nodal points. For an $\Omega_n^{1,m} \subseteq \Delta^{1,m}$, having higher degree approximation solution, can be written in terms of a linear basis as well as higher order hierarchical basis functions. Denoting the solution in $\Omega_n^{1,m}$ element by

$$\mathcal{G}(x) = \bar{U}_{1,n}^{1,m,\bar{p}}(x, t_1), \quad x \in \Omega_n^{1,m} = [x_n^{1,m}, x_{n+1}^{1,m}] \quad (4.23a)$$

We set out to find 'M' new grid points

$$a_1, a_2, a_3, a_4, \dots, a_j, \dots, a_{M-1}, a_M \\ (x_n^{1,m} \leq a_j \leq x_{n+1}^{1,m}); \quad (1 \leq j \leq M)$$

and corresponding piece-wise linear degree polynomials $\mathcal{H}_j(x)$ ($0 \leq j \leq M$) defined on these subintervals

$$[x_n^{1,m}, a_1], [a_1, a_2], \dots, [a_{M-1}, a_M], [a_M, x_{n+1}^{1,m}]$$

such that the polynomial $\mathcal{G}(x)$, is uniformly

approximated by piece-wise polynomials $\mathcal{H}_j(x)$

($0 \leq j \leq M$) on $\Omega_n^{1,m}$. That is

$$|\mathcal{G}(x) - \mathcal{H}_j(x)| \leq \varepsilon \quad \forall x \in [a_j, a_{j+1}] \quad \text{and } 0 \leq j \leq M$$

$$a_0 = x_n^{1,m} \quad \text{and} \quad a_{M+1} = x_{n+1}^{1,m}. \quad (4.23b)$$

In order to obtain the new grid points satisfying the equation (4.23b), we proceed as follows :

- (i). First, construct the straight line $\mathcal{L}(x)$ passing through $(x_n^{1,m}, \mathcal{G}(x_n^{1,m}))$ and $(x_{n+1}^{1,m}, \mathcal{G}(x_{n+1}^{1,m}))$ given by

$$\mathcal{L}(x) = \mathcal{G}(x_n^{1,m}) + (x - x_n^{1,m}) \frac{\mathcal{G}(x_{n+1}^{1,m}) - \mathcal{G}(x_n^{1,m})}{x_{n+1}^{1,m} - x_n^{1,m}}$$

$$x \in \Omega_n^{1,m}$$

(4.23c)

- (ii). Determine the stationary points of $\mathcal{G}(x) - \mathcal{L}(x)$ for

$$x \in \Omega_n^{1,m}$$

find $x^* \in \Omega_n^{1,m}$ such that

$$\frac{\partial}{\partial x} (\mathcal{G}(x^*) - \mathcal{L}(x^*)) = 0 = W(x^*) \quad (\text{say})$$

The resulting equation has the form

$$\sum_{j=1}^{N_n^{1,m}} \bar{U}_{1,j}^{1,m}[t] \frac{\partial}{\partial x} \psi_{1,j}^{1,m}(x) + \sum_{j'=1}^{N_h^{1,m}} \alpha_{1,j'}^{1,m} \frac{\partial}{\partial x} \bar{\phi}_{1,j'}(x) - \frac{\mathcal{G}(x_{n+1}^{1,m}) - \mathcal{G}(x_n^{1,m})}{x_{n+1}^{1,m} - x_n^{1,m}} = 0$$

(4.23d)

Note that required grid points are nothing but the zeros of the function $W(x) = 0$. Since $\mathcal{G}(x)$ is a polynomial $W(x)$ is also a polynomial, with real coefficients and by Rolle's theorem there exists at least one real root of $W(x^*) = 0$ which lies between $x_n^{1,m}$ and $x_{n+1}^{1,m}$ respectively. If $\Omega_n^{1,m} \in \mathcal{P}_5$, we shall restrict ourselves to only n^* roots of $W(x^*) = 0$ depending on the degree of finite element approximation used in $\Omega_n^{1,m}$ as given in following table (4.2).

| | | | | |
|--------------------------------|---|---|---|---|
| $n^* =$ | 1 | 1 | 2 | 2 |
| $\text{Deg}(\Omega_n^{i,m}) =$ | 2 | 3 | 4 | 5 |

Table 4.2a Selection of grid points in an element of p to h conversion

(iii). The choice of these n^* grid points is done as follows : Let $y_1, y_2, \dots, y_{n'-1}, y_{n'}$, ($n' \geq 1$) be the calculated roots of $W(x^*)$ which lie between $x_n^{i,m}$ and $x_{n+1}^{i,m}$ of $\Omega_n^{i,m}$.

(a). If $\underline{n'} = \underline{n^*}$: all of these roots are taken as n^* grid points in the element $\Omega_n^{i,m}$

(b). If $\underline{n'} > \underline{n^*}$: Compute $|\mathcal{G}(y_j) - \mathcal{L}(y_j)|$ for $j = 1, 2, \dots, n'$ and choose the first n^* according to the values of $|\mathcal{G}(y_j) - \mathcal{L}(y_j)|$ in descending order.

(c). If $\underline{n'} < \underline{n^*}$: All the calculations in the step 2 of mesh refinement strategy are repeated over the newly formed subintervals

$$[x_n^{i,m}, y_1], [y_1, y_2], \dots, [y_{n'-1}, y_{n'}], [y_{n'}, x_{n+1}^{i,m}]$$

Now collect all the new roots and select the required roots (n^*) according to step (b).

(iv). If $y_1, y_2, \dots, y_{n^*-1}, y_{n^*}$ ($n^* \geq 1$) be the required (n^*) grid points which lie between $x_n^{i,m}$ and $x_{n+1}^{i,m}$ of $\Omega_n^{i,m}$, construct the piece-wise linear interpolation polynomials $\mathcal{H}_j(x)$ ($0 \leq j \leq n^*$) defined on these subintervals

$$[y_0, y_1], [y_1, y_2], \dots, [y_{n^*-1}, y_{n^*}], [y_{n^*}, y_{n+1}^*]$$

where $y_0 = x_n^{i,m}$ and $y_{n^*+1} = x_{n+1}^{i,m}$.

(v). These piece-wise linear polynomials $\mathcal{H}_j(x)$ ($0 \leq j \leq n^*$)

represent polynomial $\mathcal{G}(x)$, $x \in \Omega_n^{1,m}$ closely. These steps described above for p-h conversion, have been shown schematically in Fig. 4.6b.

The important features of the p to h conversion procedure can be summarized as follows. In order to derive maximum benefit of the earlier p version calculations, a gradient based non-uniform h refinement of the highest order elements has been implemented. Since the gradient is the driving force in the present applied problem, the region of large error in the high gradient region of the physical problem is taken care of. Further, the conversion of p to h version offers gain in the distribution of grid points. At the same time, the degree of FE approximation has been converted to piece wise linear degree and then is a consequent increase in the dimension of associated solution space.

Construction of initial condition on the refined mesh at $t = t_1$:

Let $\bar{U}_1^{1,m,\bar{p}}(x,t) \in \mathcal{S}^{1,m,\bar{p}}$ be the approximate FE solution of p-version on $\Delta^{1,m}$. Then the solution $U^{1,m,\bar{p}}(x,t)$ can be expressed as a linear combination of linear and hierarchical basis functions according to the equation (4.22) with the coefficients $\bar{U}_{1,j}^{1,m}[t]$ ($1 \leq j \leq N_n^{1,m}$) and $\alpha_{1,j}^{1,m}[t]$ ($1 \leq j \leq N_h^{1,m}$) which correspond to the solution values at the nodal points and the additional freedom.

Initial condition : p refinement

It is observed noted that the meshes $\Omega^{1,m}$ and $\Omega^{1,m+1}$ are equal but the degree of approximation may vary in certain elements

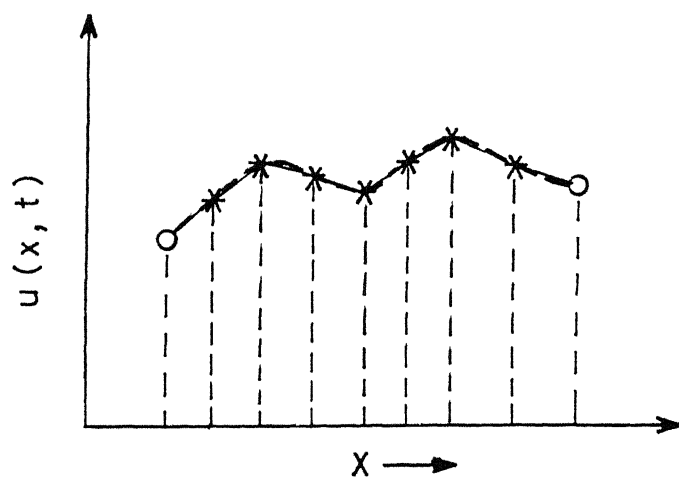
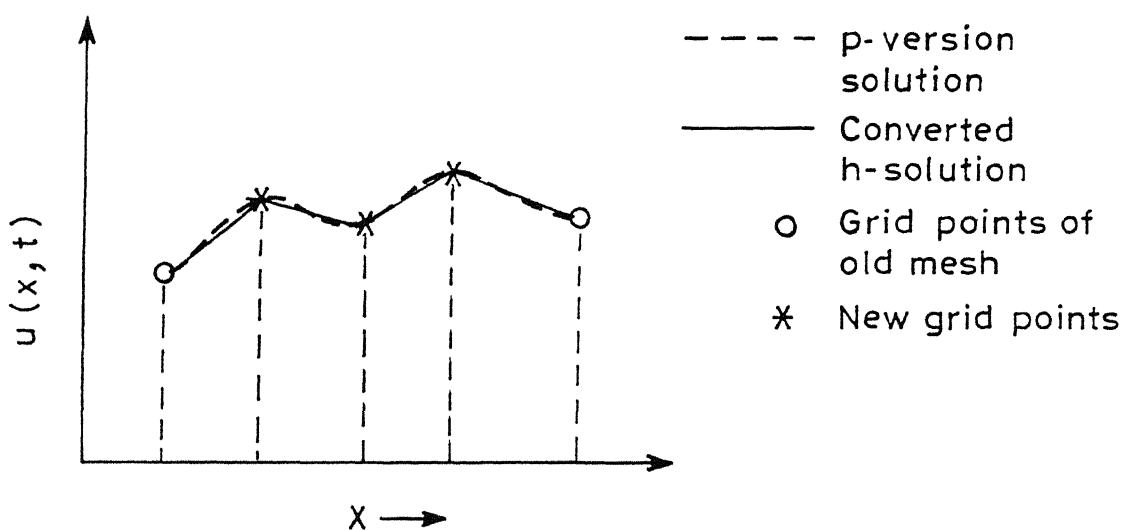
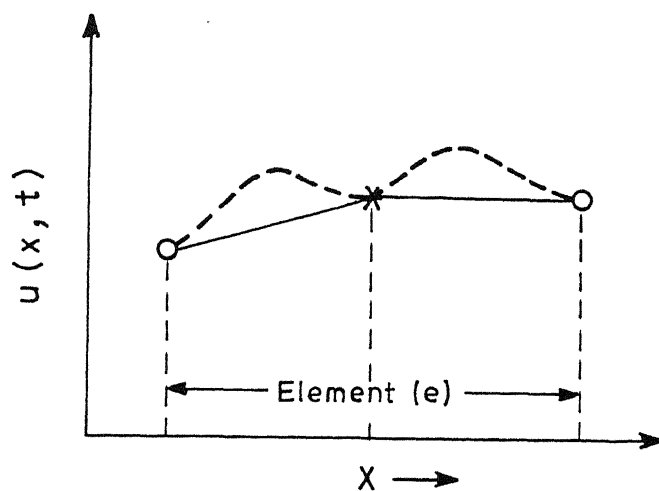


Fig. 4.6 b APPROXIMATION OF COMPUTED FE SOLUTION DURING p TO h CONVERSION IN A HIGHER ORDER ELEMENT .

from m^{th} to $(m+1)^{th}$ adaption. The procedure for the h version algorithm is implemented for obtaining the initial condition for nodal variables of the approximate solution on the refined mesh $\Omega^{i,m+1}$. Further, the additional unknown variables $(\alpha_{1,j}^{i,m+1}[t])$'s) which have been introduced due to p refinement, are taken to be zero for the approximation of initial condition on the refined mesh $\Delta^{i,m+1}$. This means that we have considered only the linear degree approximation of FE solution space $\mathcal{S}^{i,m+1,\bar{u}}$ for the approximation of initial condition $\bar{u}_1^{i,m+1,\bar{u}}(x,t)$. Thus

$$\mathcal{S}^{i,m,\bar{u}} = \mathcal{S}^{i,m+1,\bar{u}} \quad \forall m = 1, \dots, M$$

and

$$\dim (\mathcal{S}^{i,m+1,\bar{u}}) = \mathcal{N}_n^{i,m+1} + \mathcal{N}_h^{i,m+1} = \mathcal{N}_d^{i,m+1}$$

where $\mathcal{N}_h^{i,m+1} = 0$.

h-p version :

Since the p version elements are converted to h version linear elements the recently computed solution on $\Delta^{i,m}$ is interpolated over the refined mesh $\Delta^{i,m+1}$ to get an starting guess solution at $t = t_1$ for $(m+1)^{th}$ adaption. Further, the purpose of, de-refinement, the starting guess solution is determined in a similar way as explained in the h version algorithm .

i.e., for any $\tilde{u} \in \Omega_n^{i,m+1} \subseteq \Delta^{i,m+1}$, we have

$$\tilde{u} = \tilde{x} \in \Omega_n^{i,m} \subseteq \Omega^{i,m} \quad \text{for some } n' \quad (1 \leq n' \leq \mathcal{N}_e^{i,m}) .$$

same procedure for the determination of initial condition on the initial mesh $\Delta^{1+1,1}$ at $t = t_1$ as in h version. Thus we have,

$$\bar{U}^{1+1,1,\bar{U}}(x,t) = \bar{U}^{1,M,\bar{U}}(x,t) \in \mathcal{S}^{1+1,1,\bar{U}}$$

On the other hand, if

$$\bar{U}^{1,M,\bar{P}}(x,t) \in \mathcal{S}^{1,M,\bar{P}} \text{ where } \mathcal{S}^{1,M,\bar{P}} \text{ is a}$$

finite dimensional solution space of higher degree then for the elements of higher degree $\Omega_1^{1,M}$ ($1 \leq i \leq N^{1,M}$) of $\Delta^{1,M}$ the p to h version conversion is adopted and the solution is computed at the refined grid points by the interpolation of the solution

$$\bar{U}^{1,M,\bar{P}}(x,t) \in \mathcal{S}^{1,M,\bar{P}}.$$

Thus we obtain

$$\bar{U}^{1+1,1,\bar{P}}(x,t) \in \mathcal{S}^{1+1,1,\bar{P}}$$

the initial condition at $t = t_1$ on the coarse mesh $\Omega^{1+1,1}$ of linear degree approximation. Naturally, the dimension of the solution space $\mathcal{S}^{1+1,1,\bar{P}}$ changes at various points whenever the computed solution with p version is obtained at the previous time interval. The initial condition for the refined mesh is chosen as the starting guess solution for the same mesh which is useful for the solution of the non-linear problem.

4.6.3. Computational procedure :

In this section, we summarize the important computational aspects of the adaptive h-p version procedure for the test problem and the flame propagation problem.

The concept of hierarchical finite element basis functions has

been used for obtaining mixed order interpolation in the h-p version. Linear basis functions are first used on the elements of the initial mesh for each time step and then the order of polynomial approximation is increased (up to a maximum degree of five) in the regions of the mesh where the error is large. The identification of these regions is done by the statistical procedure. This leads to the introduction of higher order nodes and additional degrees of freedom for such new modes in some elements of the coarse mesh, although the number of elements remains unaltered. Thus, when the new degrees of freedom are introduced, the total number of unknown variables in the finite element approximation is increased.

The maximum order of polynomial approximation has been set equal to five in the present work. Very high order polynomials are susceptible to unrealistic numerical oscillations and it was observed during computations that the quality of the solution deteriorated beyond $p = 5$. At each stage of adaption, the error indicators have been re-computed to test the error criterion.

Since the refinement is made hierarchically, some of the matrices arising in the discretized form of differential equations (4.4) on the coarse mesh are preserved. This is a very useful feature which results in saving a lot of computational effort, while calculating the finite element solution. Further, the hierarchical degrees of freedom appear as perturbations on the original solution rather than its substitutes. So, the resultant matrix equations of the associated equation (4.4.) has a more diagonally dominant form than that obtainable in a direct approximation involving an identical number of non-hierarchic

degrees of freedom. This feature is very much useful in accelerating the convergence of the iterative method employed for the resulting matrix equations.

During the p version adaption, the computed initial solution on the coarse mesh with linear degree approximation has been used as the initial solution on the adapted mesh (with higher degrees of freedom). Further, when the p version is converted into h version in the h-p refinement procedure, the new starting guess solution has been constructed with piece-wise linear degree of approximation on each element of the refined mesh. This is achieved by converting the local higher order approximation in any element to a piece-wise linear approximation on the newly formed sub-elements. In fact, the newly introduced nodes have been chosen in such a way that the piece-wise linear degree approximation closely represents the higher order approximation available on the master element. Now, the solution on the previous mesh is interpolated to obtain the approximate solution at the newly introduced nodes of the refined mesh. The initial condition at a given time instant has also been calculated by a similar procedure whenever the final computed solution at previous time is obtained by the h-p refinement scheme.

In the p refinement scheme, the degree of the approximation in the elements with large elemental local error indicators is increased by one and the solution is recomputed at the same time. Thus, the elements in the refinement region become hierarchic of a certain degree, whereas elements outside the refinement region would be hierarchic but of lower degree. The process is continued till the maximum prescribed degree of approximation is attained.

At this stage, if further adaption is needed for the satisfaction of the error criterion by the *a posteriori* error, we first classify the elemental error indicators according to classification of adaption sets as described in the section (4.6.2). Thus, the respective regions are identified and the refinement is performed. The tolerance used for the h-p refinement are the same as that of used for h-refinement.

4.7. Grid speed adaptive strategy :

4.7.1. Description of grid speed technique :

Several authors (Miller and Miller, 1981a,b; Dwyer and Sanders, 1983a; Larrousturou, 1986; Dervieux et al., 1989; Benkhaldoun et al., 1988b; Ramos, 1985), have worked on moving grid methods and algorithms for the determination of the nodal velocities in the adaptive grid procedures. The concept of moving all the mesh points in the same direction at each time step has been worked out by Larrousturou and co-authors (Larrousturou, 1986; Dervieux et al., 1989; Benkhaldoun et. al, 1988b) for the front propagation problems. In their work, the grid velocity $C(t)$ is chosen in such a way that the integral of the solution variable on the computational domain remains constant. Ramos (1985) have calculated the flame speed using the maximum temperature gradient in the adaptive finite element method.

In the following section, the development of a different and simple grid speed strategy has been discussed and the behaviour of various error measures during the application of this strategy to the test problem have been presented. In contrast to the previous works, here the differential equation for temperature has been

used to calculate the flame speed after invoking the assumption of uniform front velocity.

4.7.2. Description of the algorithm :

In the test problem, the solution profile at each time instant consists of a narrow wave front region of large gradients and a wide zone where the solution is constant. As the wave front moves, the previously adapted grid lay out in the front region also has to move at the speed of the front in order to provide optimal spacing for later times. In the results presented earlier for h or h-p version schemes, error measures were checked for each time step and grid-adaptions were performed whenever necessary. This results in a large computational effort for the grid adaption process. A computationally inexpensive way of achieving grid movement would be to calculate the front speed for each instant of time (from the computed solution profile) and displace the grid points at the speed of the front. This idea is explained as follows.

Consider a mesh with a fixed number of grid points, which are moving at each time step, i.e, the nodal locations x are functions of time. On a moving grid, the time derivative of any variable at a grid point can be expressed as :

$$\frac{d}{dt} U(x(t),t) = \frac{\partial U}{\partial x} \frac{dx}{dt} + \frac{\partial U}{\partial t}$$

where the nodal velocity $\frac{dx}{dt}$ appears explicitly. In many situations, the solution profile does not change with respect to time and the same profile translates from one location to another. For such a case, in order to maintain the same relative position of a grid point within the front, the required grid velocity can be

calculated from the equation,

$$0 = \frac{d}{dt} U(x(t), t) = \frac{\partial U}{\partial x} \frac{dx}{dt} + \frac{\partial U}{\partial t}$$

The above equation implies that there is no change in the solution with respect to time, for a moving grid point. Now for the test problem, substituting for $\frac{\partial U}{\partial t}$ from eqs. (4.1a), with $a(x) = 1$, and $b(x) = 0$, the grid velocity $C(t)$ is obtained as :

$$C(t) = \frac{dx}{dt} = - \left(f(x, t) + \frac{\partial^2 U}{\partial x^2} \right) / \frac{\partial U}{\partial x} \quad (4.25a)$$

The resulting new location for node j at $t = t_{i+1}$ is given as follows :

$$x_{j+1}(t_{i+1}) = x_j(t_i) + C(t_i)(t_{i+1} - t_i). \quad (4.25b)$$

The important steps involved in the grid speed strategy are explained below algorithmically. The corresponding flow chart is shown in the Fig. 4.7.

Step 1 : Calculate the finite element solution at $t = t_i$:

Calculate the FE solution at $t = t_i$ on the mesh $\Delta^{i,m}$ with given initial condition and the starting guess solution $t = t_i$.

Step 2 : Calculate the Grid Velocity :

Calculate time derivative of computed solution at $t = t_i$ and spatial derivative of computed solution at $t = t_i$
Calculate uniform grid speed $(C(t)) = - \frac{\partial U}{\partial t} / \frac{\partial U}{\partial x}$

Step 3 : Generate New mesh at $t = t_i$ with grid speed .

Calculate new mesh location of a node ' j ' at $t = t_i$ using $x_{j+1}(t_{i+1}) = x_j(t_i) + C(t_i)(t_{i+1} - t_i)$.

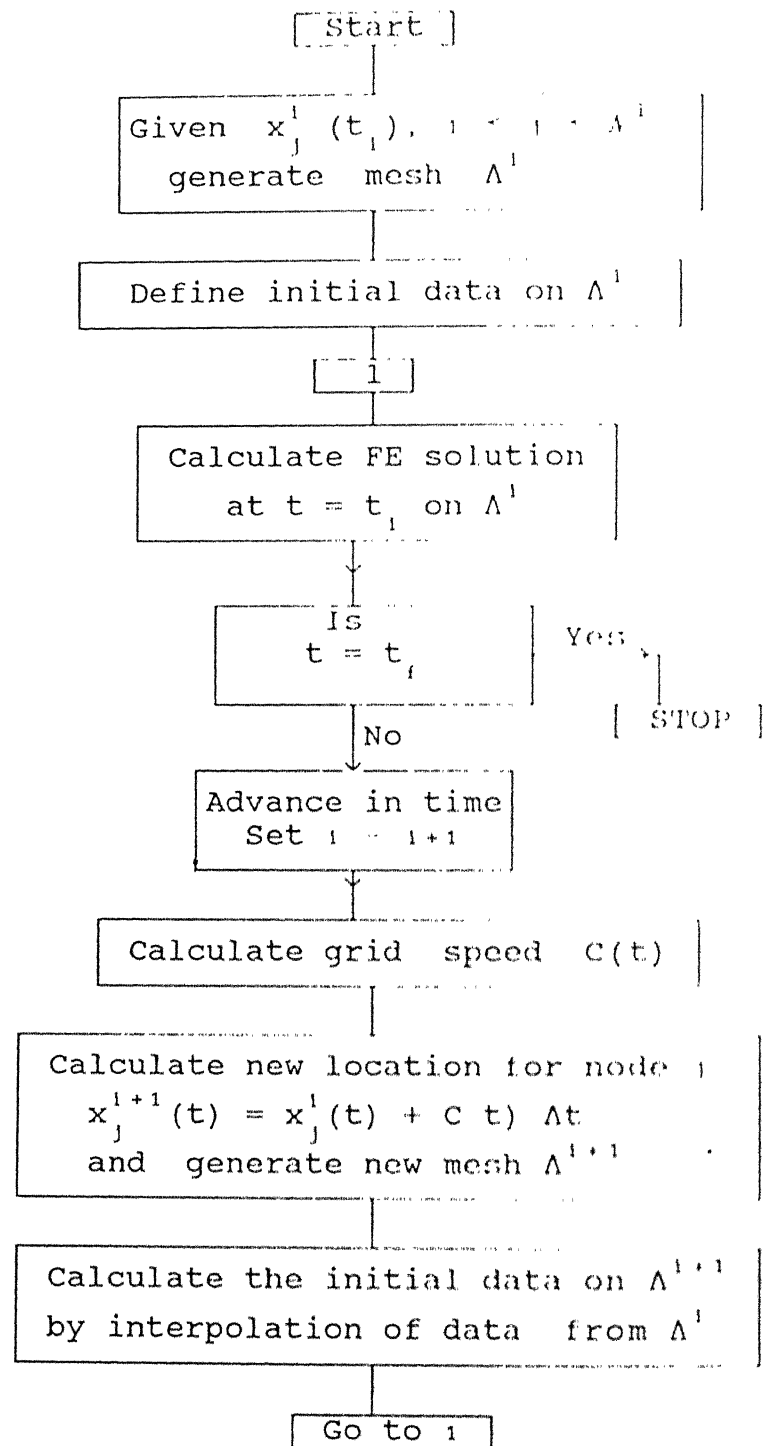


Fig. 4.7. flow chart for Grid speed strategy

Step 4 : Generate initial data on the new mesh Δ^{i+1} at $t = t_{i+1}$
 generate the initial data on the Δ^{i+1} by interpolation
 of solution on Δ^i to Δ^{i+1} , increase the time counter
 by one and go to step 1.

4.7.3. Computational procedure

The computational implementation of the grid speed strategy for the test problem as well as the flame problem is same. According to this strategy, all the nodes (except those on boundary) have been provided the same velocity $C(t)$ at a given time t . This velocity $C(t)$ is calculated as the average of all the velocities at individual grid points. Computational experience shows that the average grid velocity in the wave front zone is very close to the exact grid speed (= front speed) for the test problem. It is also observed that as the grid points move with same velocity, collapse and stretching of elements occur near the boundary which affects the solutions. In order to avoid this, special treatment of grid movement is necessary near the domain boundaries.

In the computational experiments, the boundary nodes of the domain have been locked and the interior nodes are allowed to move according to uniform grid speed. After a few time iterations, the nodes which are near to the boundary may shoot outside the boundary. So these which cross the boundary are reintroduced at the beginning of the mesh. Although, it is not necessary to reintroduce these nodes in the present problem, for more general cases it may be necessary.

It is important to note that the oscillatory nature of the

solution profile may affect the evaluation of the gradient. Further, the second derivative of the computed solution has to be evaluated carefully as it has significant contribution in the evaluation of the grid speed in wave front region. Thus, we have evaluated the first and second derivatives of the gradient by finite difference expressions which are second order accurate. The consideration of second derivative term in the evaluation of grid speed introduces some errors because of the choice of linear degree approximation employed in the finite element solution procedure. However, for second order accurate procedures, the effect of the linear degree FE approximation is negligibly small.

4.8. Results and discussion :

4.8.1. Application of adaptive strategies to Test problem :

In this section, results of solution profiles, error estimates, and grid - distributions are presented for the flame propagation as well as the test problems. The test problem has been considered in order to analyze the effectiveness of the error measures used for grid - adaptations in both the h version and hp version methods. In fact, since the exact solution is available for the test problem, the true gradient errors and relative errors of the computed solutions can also be estimated. In view of the sensitivity of error estimates to the smoothness of the solution, a local smoothening procedure using piece-wise cubic spline interpolation has been attempted in h version for improving the estimates of the true gradient error. Detailed comparisons of the relative performance of h and hp version methods have been presented. Finally, applying the grid - velocity concept to

h-version grid adaption, the construction of computationally inexpensive meshes has been attempted and the behaviour of a *posteriori* error on such meshes has been studied.

(I). Uniform mesh results :

Bieterman and Babuska, (1982a-b) have shown the equivalence of the various error measures (see, section 3.4, Chapter 3) for linear problems. Here, for validating the computational results of the present analysis comparisons of the computed and the exact solutions have been provided graphically. The characteristics of different error estimators have also been plotted. The results have been presented in the tabular form (table 4.3a-d) and the comparisons with the results of Bieterman and Babuska (1982a) have also been shown at closely approximate time values (table 4.3c-d). Further, the notations of the various quantities used in the subsequent tables have been described in the table 4.2b.

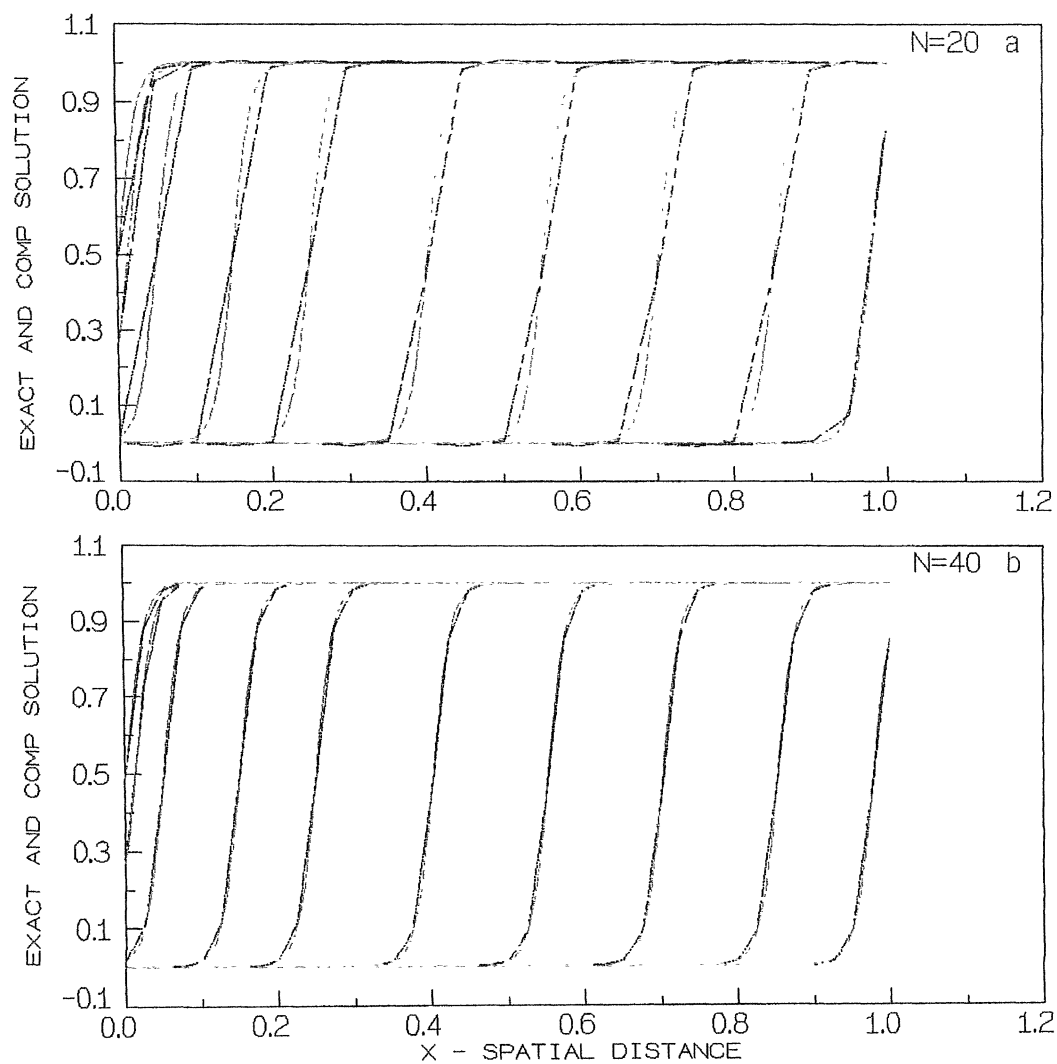
The test problem was solved on a sequence of uniform meshes with 20, 40, 80 and 160 elements. The reason for considering the uniform meshes is that in adaptive grid problems, additional error may result if the movement of grid points is not synchronized with that of the front. Therefore, in order to analyze the nature of the error estimates without moving-grid effects, uniform meshes have been considered prior to adaptive grid solutions.

(a). Comparison of exact and computed solution profiles :

The exact and the calculated solution profiles obtained using uniform meshes for the test problem are shown in Figs. 4.8a-d, for different time instants. For all the meshes considered, the

| Variable | Description |
|--------------|--------------------------------------|
| Rem (Rm) | Number of elements removed |
| Ref (Rf) | Number of elements refined |
| Stf | Nuber of elements stufed |
| Adp | Adaption number |
| N (Eles) | Number of elements considered. |
| Nodes(Nos) | Number of nodes in the domain |
| Dof | Total number of degrees of freedom. |
| Deg | Maximum degree of approximation used |
| Apos. err | Aposteriori error |
| com.grd | Computational gradient |
| Grd.err | Gradient true error |
| Eff.ind | Effectivity index |
| Non-lin-ter | Non-linear iterations |
| α -st | α -strategy |
| comp sol | Computed solution |
| Exact sol | Exact solution |
| h | h-version |
| hp | hp-version |

Table 4.2b. Description of the variables used in subsequent tables.



| Solution | Line_type |
|------------|-----------|
| Comp soln | ----- |
| Exact soln | ----- |

| Adp | Time | Eles | Nods |
|-----|----------|------|------|
| 1 | 0.000050 | 20 | 21 |
| 2 | 0.001250 | 20 | 21 |
| 3 | 0.005000 | 20 | 21 |
| 4 | 0.015000 | 20 | 21 |
| 5 | 0.025000 | 20 | 21 |
| 6 | 0.040300 | 20 | 21 |
| 7 | 0.055300 | 20 | 21 |
| 8 | 0.070300 | 20 | 21 |
| 9 | 0.085300 | 20 | 21 |
| 10 | 0.097800 | 20 | 21 |

| Adp | Time | Eles | Dof |
|-----|----------|------|-----|
| 1 | 0.000050 | 40 | 41 |
| 2 | 0.001250 | 40 | 41 |
| 3 | 0.005000 | 40 | 41 |
| 4 | 0.015000 | 40 | 41 |
| 5 | 0.025000 | 40 | 41 |
| 6 | 0.040300 | 40 | 41 |
| 7 | 0.055300 | 40 | 41 |
| 8 | 0.070300 | 40 | 41 |
| 9 | 0.085300 | 40 | 41 |
| 10 | 0.097800 | 40 | 41 |

Fig. 4.8 TEST PROBLEM (Uniform Mesh) : SOLUTION PROFILE

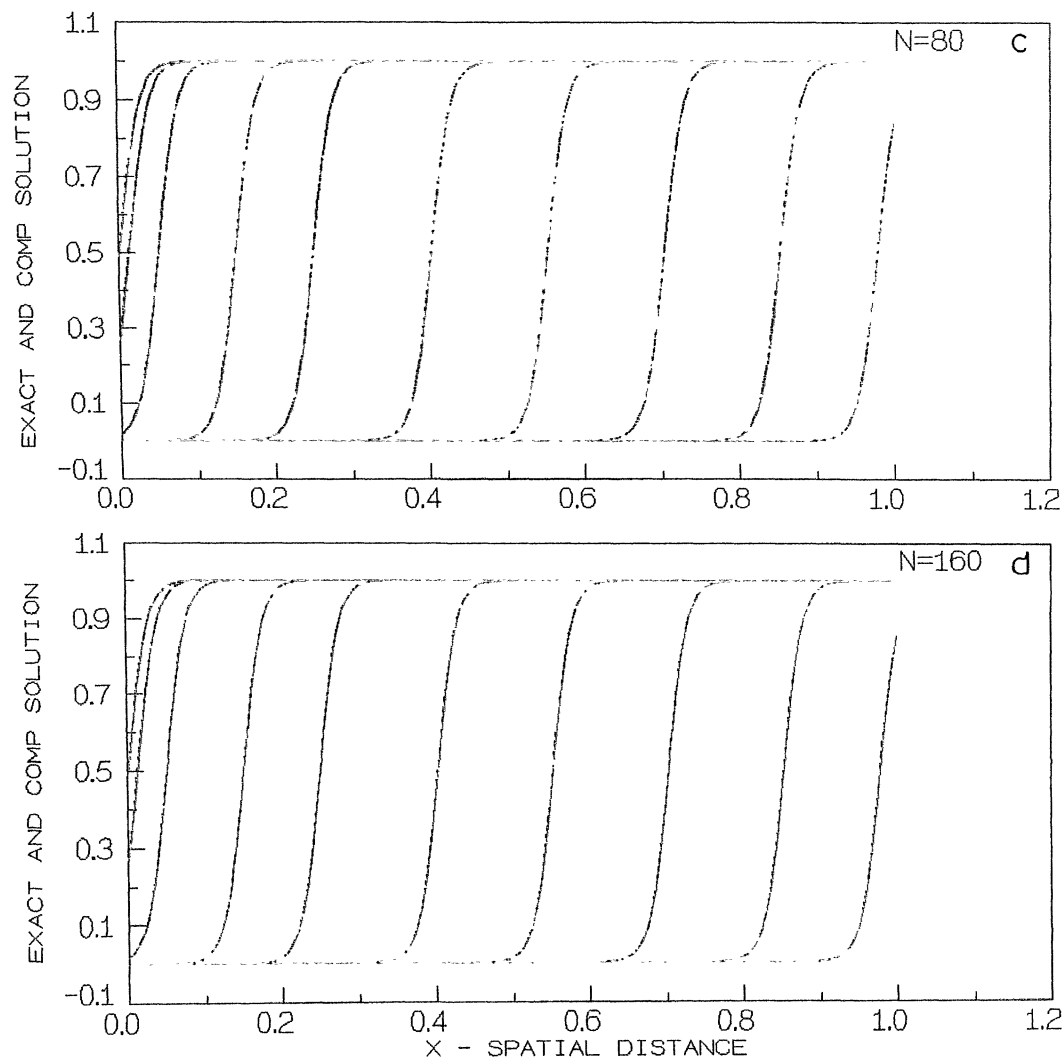


Fig. 4.8 TEST PROBLEM (Uniform Mesh) : SOLUTION PROFILE

| Solution | Line_type |
|------------|-----------|
| Comp soln | ————— |
| Exact soln | - - - - - |

| Adp | Time | Eles | Nods |
|-----|-----------|------|------|
| 1 | 0 .000050 | 80 | 81 |
| 2 | 0 .001250 | 80 | 81 |
| 3 | 0 .005000 | 80 | 81 |
| 4 | 0 .015000 | 80 | 81 |
| 5 | 0 .025000 | 80 | 81 |
| 6 | 0 .040300 | 80 | 81 |
| 7 | 0 .055300 | 80 | 81 |
| 8 | 0 .070300 | 80 | 81 |
| 9 | 0 .085300 | 80 | 81 |
| 10 | 0 .097800 | 80 | 81 |

| Adp | Time | Eles | Dof |
|-----|-----------|------|-----|
| 1 | 0 .000050 | 160 | 161 |
| 2 | 0 .001250 | 160 | 161 |
| 3 | 0 .005000 | 160 | 161 |
| 4 | 0 .015000 | 160 | 161 |
| 5 | 0 .025000 | 160 | 161 |
| 6 | 0 .040300 | 160 | 161 |
| 7 | 0 .055300 | 160 | 161 |
| 8 | 0 .070300 | 160 | 161 |
| 9 | 0 .085300 | 160 | 161 |
| 10 | 0 .097800 | 160 | 161 |

| Time | Elements | | | |
|----------|----------|--------|--------|---------|
| | N = 20 | N = 40 | N = 80 | N = 160 |
| 0.000050 | 1.3396 | 0.6598 | 0.3323 | 0.1662 |
| 0.005150 | 1.8644 | 0.9412 | 0.4700 | 0.2348 |
| 0.010250 | 1.8785 | 0.9419 | 0.4708 | 0.2352 |
| 0.023600 | 1.9847 | 0.9423 | 0.4708 | 0.2352 |
| 0.032050 | 1.8534 | 0.9496 | 0.4708 | 0.2352 |
| 0.046550 | 1.8171 | 0.9479 | 0.4708 | 0.2352 |
| 0.056050 | 1.8126 | 0.9417 | 0.4708 | 0.2352 |
| 0.060050 | 1.8990 | 0.9444 | 0.4708 | 0.2352 |
| 0.070050 | 1.8990 | 0.9444 | 0.4708 | 0.2352 |
| 0.088050 | 1.9540 | 0.9396 | 0.4708 | 0.2352 |
| 0.099800 | 1.3472 | 0.6760 | 0.3335 | 0.1664 |

Table 4.3a. Aposteriori error predictions

| Time | Elements | | | |
|----------|----------|--------|--------|---------|
| | N = 20 | N = 40 | N = 80 | N = 160 |
| 0.000050 | 1.4257 | 0.6639 | 0.3290 | 0.1661 |
| 0.005150 | 1.9995 | 0.9421 | 0.4654 | 0.2346 |
| 0.010250 | 1.9901 | 0.9382 | 0.4662 | 0.2350 |
| 0.023600 | 1.5307 | 0.8595 | 0.4662 | 0.2350 |
| 0.032050 | 1.1912 | 0.9216 | 0.4661 | 0.2350 |
| 0.046550 | 1.4355 | 0.8691 | 0.4662 | 0.2350 |
| 0.056050 | 1.7108 | 0.8620 | 0.4662 | 0.2350 |
| 0.060050 | 2.0083 | 0.9462 | 0.4662 | 0.2350 |
| 0.070050 | 2.0083 | 0.9462 | 0.4662 | 0.2350 |
| 0.088050 | 1.2401 | 0.9108 | 0.4661 | 0.2350 |
| 0.099800 | 1.4392 | 0.6479 | 0.3277 | 0.1661 |

Table 4.3b. Gradient of true error predictions

(i). Present work :

| Time | Elements | | | |
|----------|----------|--------|--------|---------|
| | N = 20 | N = 40 | N = 80 | N = 160 |
| 0.005150 | 0.5478 | 0.2581 | 0.1275 | 0.0643 |
| 0.010250 | 0.5450 | 0.2569 | 0.1277 | 0.0644 |
| 0.023600 | 0.4192 | 0.2354 | 0.1277 | 0.0644 |
| 0.032050 | 0.3262 | 0.2524 | 0.1277 | 0.0644 |
| 0.046550 | 0.3931 | 0.2380 | 0.1277 | 0.0644 |
| 0.056050 | 0.4685 | 0.2361 | 0.1277 | 0.0644 |
| 0.060050 | 0.5500 | 0.2591 | 0.1277 | 0.0644 |
| 0.070050 | 0.5500 | 0.2591 | 0.1277 | 0.0644 |
| 0.088050 | 0.3396 | 0.2494 | 0.1277 | 0.0644 |
| 0.099800 | 0.5268 | 0.2372 | 0.1200 | 0.0608 |

(ii). Results of Babuska and Rheinboldt

| Time | Elements | | | |
|---------|----------|--------|--------|---------|
| | N = 20 | N = 40 | N = 80 | N = 160 |
| 0.03200 | 0.4325 | 0.2634 | 0.1308 | 0.0649 |
| 0.05600 | 0.5697 | 0.2431 | 0.1309 | 0.0649 |
| 0.06000 | 0.6437 | 0.2818 | 0.1309 | 0.0649 |
| 0.08800 | 0.3473 | 0.2673 | 0.1309 | 0.0649 |

Table. 4.3c. Relative error predictions

(i). Present work :

| Time | Elements | | | |
|----------|----------|--------|--------|---------|
| | N = 20 | N = 40 | N = 80 | N = 160 |
| 0.005150 | 0.9324 | 0.9991 | 1.0099 | 1.0006 |
| 0.010250 | 0.9440 | 1.0038 | 1.0100 | 1.0007 |
| 0.023600 | 1.2966 | 1.0963 | 1.0100 | 1.0007 |
| 0.032050 | 1.5560 | 1.0303 | 1.0101 | 1.0007 |
| 0.046550 | 1.2658 | 1.0906 | 1.0100 | 1.0007 |
| 0.056050 | 1.0595 | 1.0924 | 1.0100 | 1.0007 |
| 0.060050 | 0.9456 | 0.9981 | 1.0100 | 1.0007 |
| 0.070050 | 0.9456 | 0.9981 | 1.0100 | 1.0007 |
| 0.088050 | 1.5758 | 1.0316 | 1.0101 | 1.0007 |
| 0.099800 | 0.9361 | 1.0434 | 1.0177 | 1.0023 |

(ii). Results of Babuska and Rheinboldt

| Time | Elements | | | |
|---------|----------|--------|--------|---------|
| | N = 20 | N = 40 | N = 80 | N = 160 |
| 0.03200 | 1.3200 | 0.9950 | 0.9890 | 0.9960 |
| 0.05600 | 0.7760 | 1.0760 | 0.9890 | 0.9960 |
| 0.06000 | 0.8060 | 0.9030 | 0.9888 | 0.9960 |
| 0.08800 | 1.6850 | 0.9610 | 0.9870 | 0.9960 |

Table. 4.3d. Effectivity ratio predictions

computed solutions correctly predict a nearly linear profile in the moving front region. However, the slopes of the computed solution profiles are not accurate for the coarse meshes ($N = 20$ and 40) because the steep gradients are not resolved accurately. Results of this study show that the shape of the computed solution for $N > 40$ coincides with that of the exact solution. In fact, for $N > 40$, the grid size is less than the approximate width of the front, which is an important requirement for accurate resolution of the gradients. The graphs of the solution profiles as well as those of the error estimators, indicate that the accumulation of spatial discretization error affects the timewise evolution also, in coarse meshes. The time discretization error itself, however, seems to be negligible for the small time steps used in the present computations.

(b). Behaviour of a posteriori error, true gradient error, relative error and effectivity index :

The results for the a posteriori errors, true gradient errors, relative errors and effectivity index on uniform meshes at different times are presented in the Figs. 4.9. and 4.10. The observed trends of the errors can be explained considering three separate stages of the front propagation, namely : the start-up transient, uniform front motion and the extinction transient. During start- up and extinction which occur near the domain boundaries, the a posteriori error is low (Fig. 4.9a). This may be attributed to the smaller contribution of source term in the governing equation and the smaller value of the front speed. In the uniform propagation stage, the a posteriori error undergoes

systematic oscillations due to the alternate under-prediction/over-prediction of the solution around the mean value, with respect to time. The average magnitude of the *a posteriori* error, however, remains constant at this stage. As the number of elements is increased, this mean value of error decreases linearly proportional to the step size h (or inversely proportional to the number of elements). Also, the oscillations in error with respect to time are eliminated.

As regards the true gradient error (Fig. 4.9b) the following observations can be made. During the uniform propagation of the front, although the average magnitude of true gradient error is constant, the amplitude of timewise oscillations is large. This implies that the variation in the computed gradients with respect to time is more severe than that of the *a posteriori* error. However, as the gradients are better represented through grid refinements, the oscillations in true gradient error vanish. An interesting trend seen is that the error in the computed gradients decreases in the sharp transition zones, although the wall gradients at the boundaries are not accurately represented for coarse meshes. Thus the minimum value of true gradient error occurs slightly away from the wall for the coarse meshes. It is also observed that for the linear test problem considered, the *a posteriori* error and true gradient error are almost of similar magnitude, although the true gradient error is slightly smaller for coarse meshes. On fine meshes, the two error estimates become exactly equal to each other.

Fig. (4.10a) shows the variation of relative error with respect to time. The behaviour of relative error is similar to

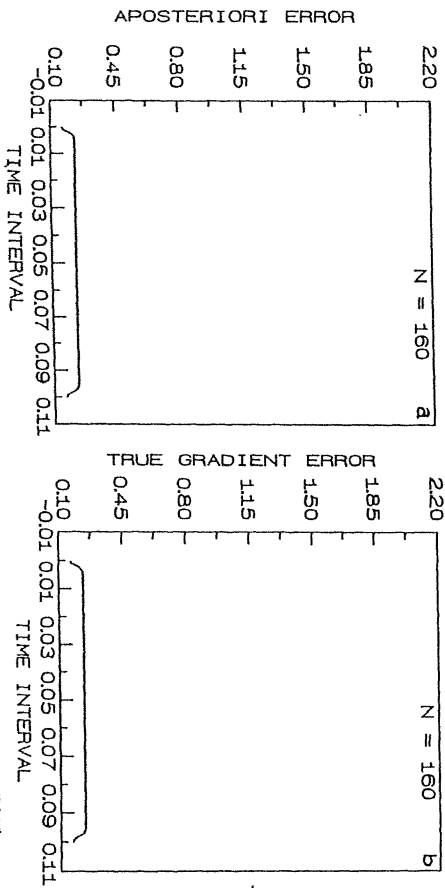
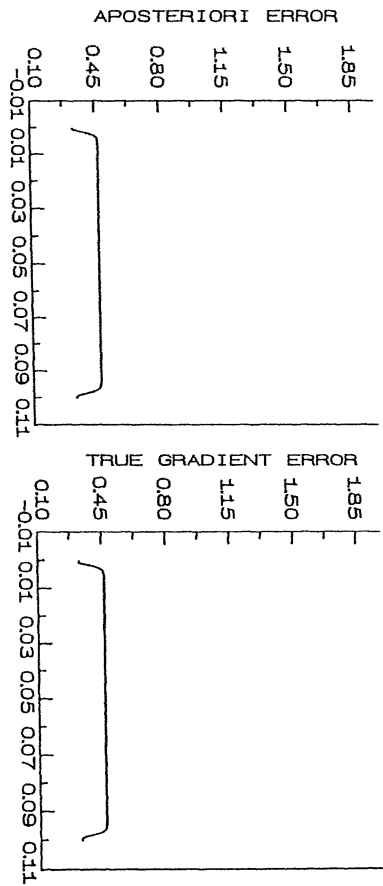
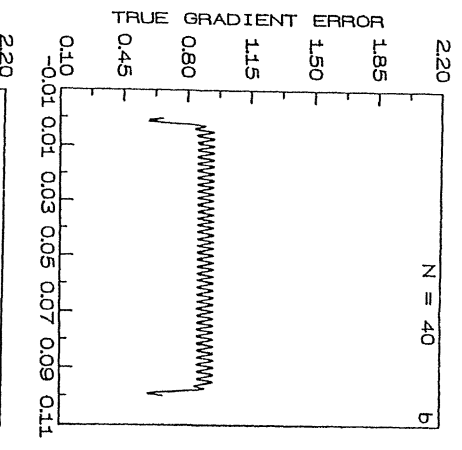
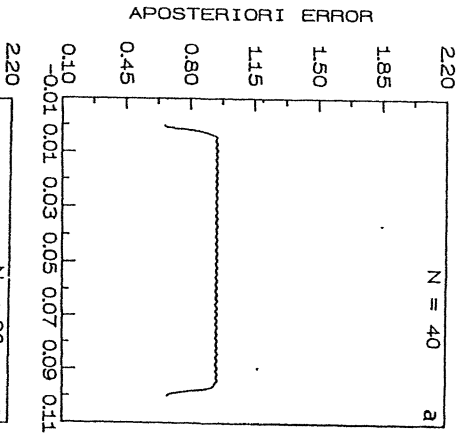
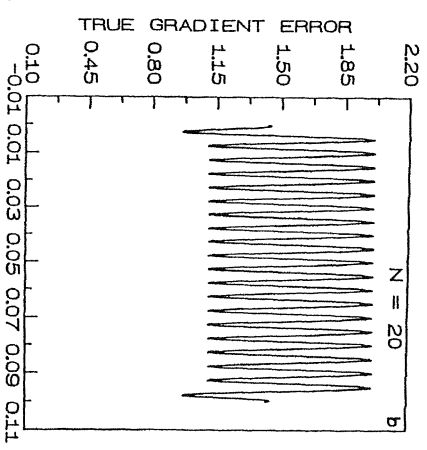
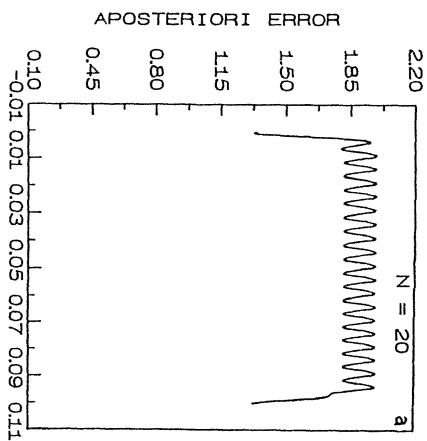


Fig. 4.9. TEST PROBLEM : TIME EVOLUTION OF ERRORS (UNIFORM MESH)
(APOSTERIORI ERROR AND TRUE GRADIENT ERROR)



N = 80

N = 80

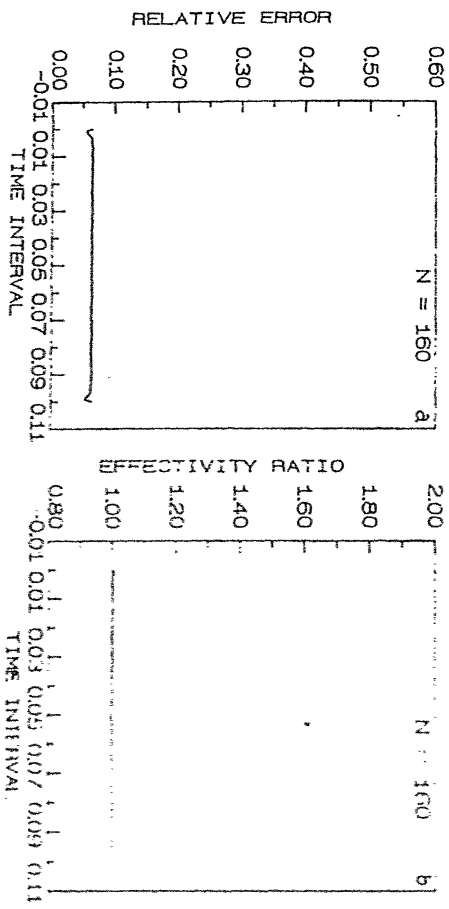
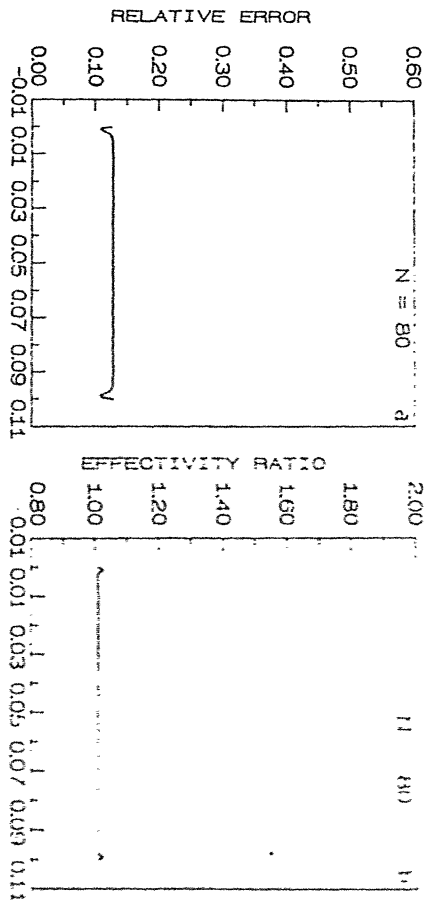
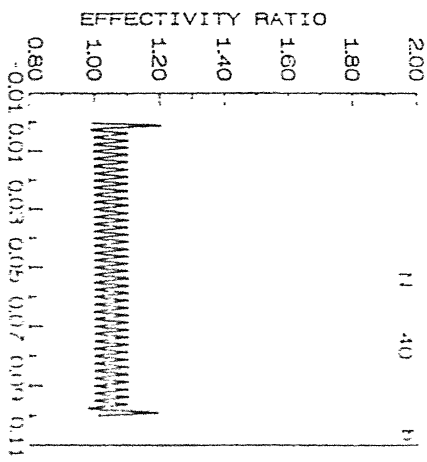
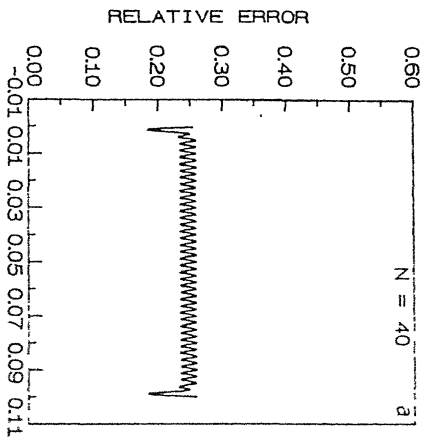
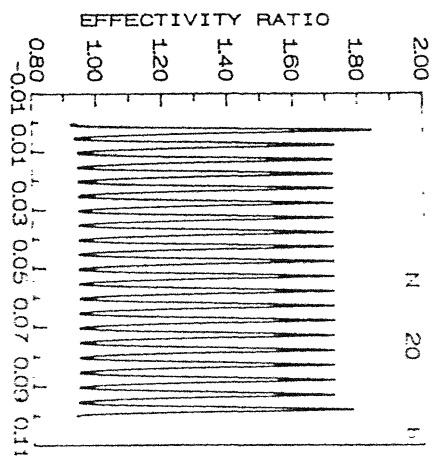
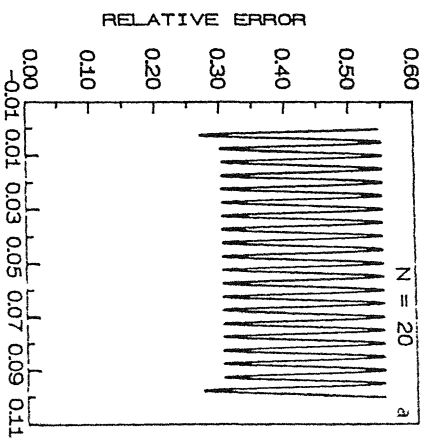


Fig. 4.10. TEST PROBLEM : TIME EVOLUTION OF ERRORS (UNIFORM MESH)
(RELATIVE ERROR AND EFFECTIVITY RATIO)



that of the true gradient error. It is observed that the relative error decreases rapidly, as the mesh becomes finer . In Fig. (4.10b), the effectivity index which is the ratio of the *a posteriori* and the true gradient errors, is plotted against time. For the coarse meshes, the effectivity index fluctuates rapidly due to fluctuations in the *a posteriori* and true gradient errors. Obviously, the oscillations of the two error measures are not in phase. As the number of elements is increased, the oscillations in the two error estimates are eliminated and the effectivity index approaches unity.

From the Figs. (4.9a-b, 4.10a-b), the following conclusions can be drawn. As the number of elements is increased, the computed solution approaches the exact solution and the *a posteriori* error closely approximates the true gradient error. Both of these errors go to zero linearly proportional to step size. Thus, for complex problems in which true gradient error can not be estimated, the *a posteriori* error may be used as an effective measure for grid adaption purposes, provided some necessary conditions on the PDE system are satisfied. Regarding the choice of the minimum number of uniform elements required for the test problem, it can be said that the step size h should be less than the front width $(2\beta)^{-1}$.

(II). Non-uniform mesh results (h and hp refinement methods) :

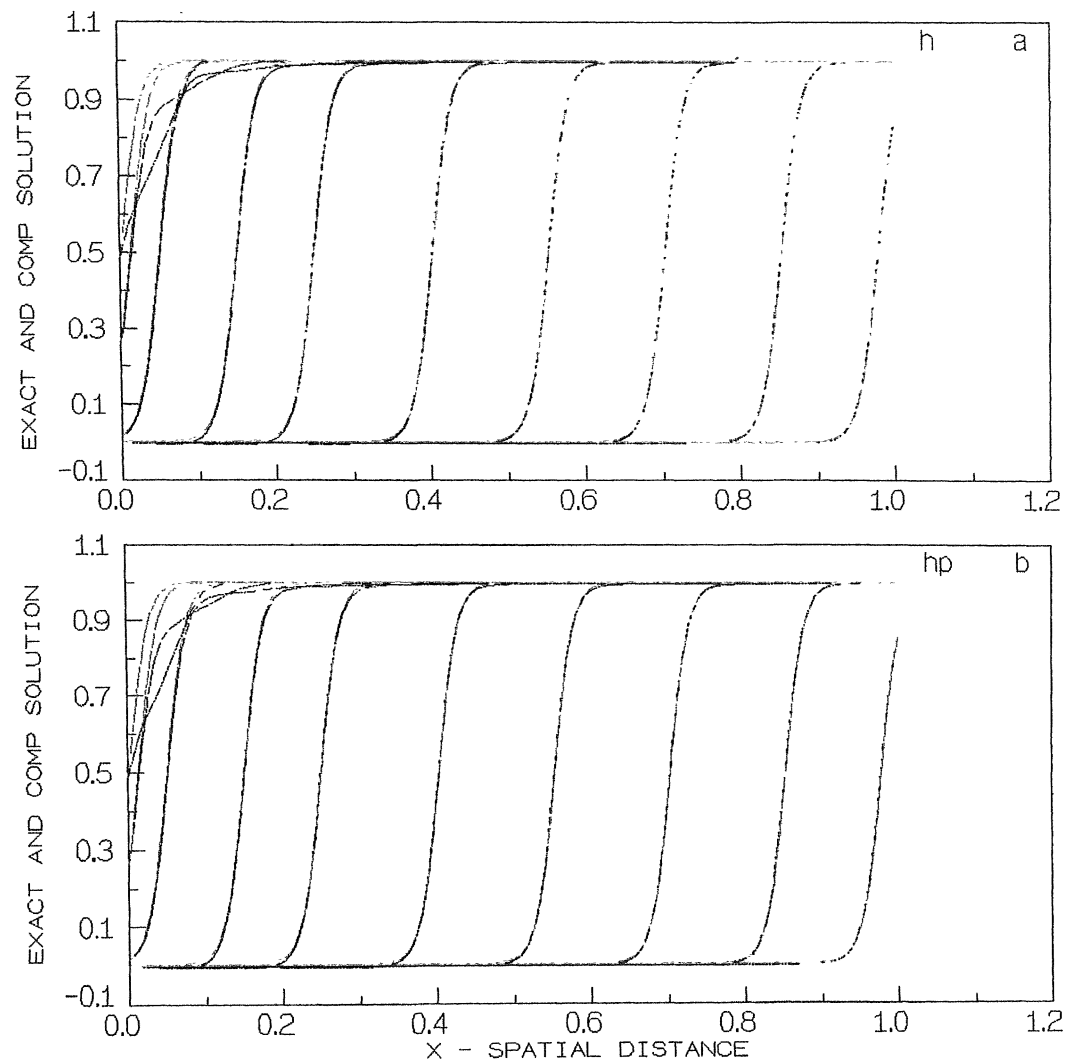
(a). Solution profiles :

The exact and the computed solution profiles for the h and hp refinement methods calculated on non-uniform meshes have been plotted in Figs. 4.11a and 4.11b. Also, the histories of grid

adaption at various time instants have been shown in the tables by the side of the figures. At each time instant, the computed solution corresponds to the final adaptive mesh on which the solution meets the error criterion. In both the methods, the adaption is carried out by starting from a grid of 10 uniform elements initially. It can be observed from the figures that the computed solution is inaccurate at initial times (near the $x = 0$ boundary) and the differences gradually disappear as the mesh adapts at various times along with the front movement. The initial deviations are obviously due to the inadequate spatial resolution by the coarse mesh. For later times, however, both the shape of the front profile and the speed of movement match exactly with the exact solution for the h and hp refinement techniques.

(b). Behaviour of a posteriori error, true gradient error, relative error and effectivity index :

The variation of the *a posteriori* error with time is depicted in Fig. 4.12a. It can be seen from the graph that the *a posteriori* errors for both the h and hp refinement methods are very high initially, indicating inadequate spatial resolution by the coarse initial mesh. But by the repeated application of the adaptive procedure, zones of large solution error are refined and the *a posteriori* error reduces rapidly and monotonically below the prescribed tolerance. As the solution front moves in time, however, the refined mesh obtained initially is not best suited. The *a posteriori* error therefore starts increasing until fresh mesh adaption begins when the error slightly exceeds the tolerance. After this adaption, the mesh is adjusted to the new



| Solution | Line_type |
|------------|-----------|
| Comp soln | ----- |
| Exact soln | |

h version mesh

| Adp | Time | Eles | Nods |
|-----|-----------|------|------|
| 1 | 0 .000050 | 19 | 20 |
| 2 | 0 .001250 | 52 | 53 |
| 3 | 0 .005000 | 119 | 120 |
| 4 | 0 .015000 | 148 | 149 |
| 5 | 0 .025000 | 159 | 160 |
| 6 | 0 .040300 | 157 | 158 |
| 7 | 0 .055300 | 136 | 137 |
| 8 | 0 .070300 | 149 | 150 |
| 9 | 0 .085300 | 147 | 148 |
| 10 | 0 .097800 | 139 | 140 |

hp version mesh

| Adp | Time | Eles | Dof |
|-----|-----------|------|-----|
| 1 | 0 .000050 | 15 | 16 |
| 2 | 0 .001250 | 55 | 56 |
| 3 | 0 .005000 | 76 | 77 |
| 4 | 0 .015000 | 139 | 140 |
| 5 | 0 .025000 | 127 | 128 |
| 6 | 0 .040300 | 139 | 140 |
| 7 | 0 .055300 | 123 | 124 |
| 8 | 0 .070300 | 121 | 147 |
| 9 | 0 .085300 | 132 | 133 |
| 10 | 0 .097800 | 98 | 99 |

Fig. 4.11 TEST PROBLEM (h & hp version) : SOLUTION PROFILE

location of the front and therefore, a *a posteriori* error is considerably less. This process of grid adjustment need not be performed at every time step, since a *a posteriori* error in many cases may be below the tolerance level. Only when the error exceeds the tolerance (in other words, the front has almost moved out of the earlier refined mesh zone), grid adaption is required. Due to this lag between front movement and grid adaption, the *a posteriori* error undergoes fluctuations with respect to time. A remarkable feature in the error analysis is that the new initial mesh at any time has been constructed with the best use of information available on the final mesh at the previous time. This reduces the number of adaptations in the total time interval.

For both h and hp refinement methods, the adaption criterion works well to keep the error below the tolerance for most of the time. In the case of hp method, the mean error is slightly larger than that in the h - version; however, the number of elements employed in the hp version is less. Further, it is observed that increasing the degree of finite element approximation in some elements locally, reduces the *a posteriori* error very slowly. Therefore, the number of adaption cycles per time step is more in the hp version in comparison to the h version. Close to extinction, the *a posteriori* error decreases for both the h and hp refinement methods, for reasons cited while discussing the results of the uniform meshes (Fig. 4.9a).

In Fig. 4.12b, the time evolutions of the true gradient error in the h and hp refinement methods are shown for the test problem. It is seen that during uniform front propagation, the mean values of the error are almost constant and equal for the h and hp

methods. However, some minute differences do exist between the true gradient error behaviours for the two methods. For instance, the amplitudes of the oscillations are smaller but the frequencies are higher, for the hp method. This is due the fact that the error decreases slowly as the order of interpolation is increased at the locations of large error; in fact, for higher order interpolations there is sometimes even an increase in the local error due to the pollution effect. Thus, more number of adaptations are often required in the hp version, which is shown in the form of higher oscillation frequency of the true gradient error. In general, the true gradient error for the hp method is larger than that for the h version.

In the light of the discussions presented above, it is important to note that for initial times the hp method does not meet the error criterion even after reaching the maximum degree of interpolation in some elements. Therefore, essentially h version adaption is carried out in the hp version until the *a posteriori* error becomes less than the tolerance. Another important feature which is apparent from the comparison of Figs. 4.12a and 4.12b is that the *a posteriori* and true gradient errors are close to each other, but exactly not equal, for the non-uniform mesh. In fact, the tolerance criterion serves as an upper limit for the *a posteriori* error while in the case of true gradient error, the mean value is closer to the tolerance.

For non-uniform meshes, the relative error behaviour is similar to that of true gradient error as seen from Fig. 4.12c. The effectivity index (Fig. 4.12d) which is the ratio between the *a posteriori* and true gradient errors, decreases steeply

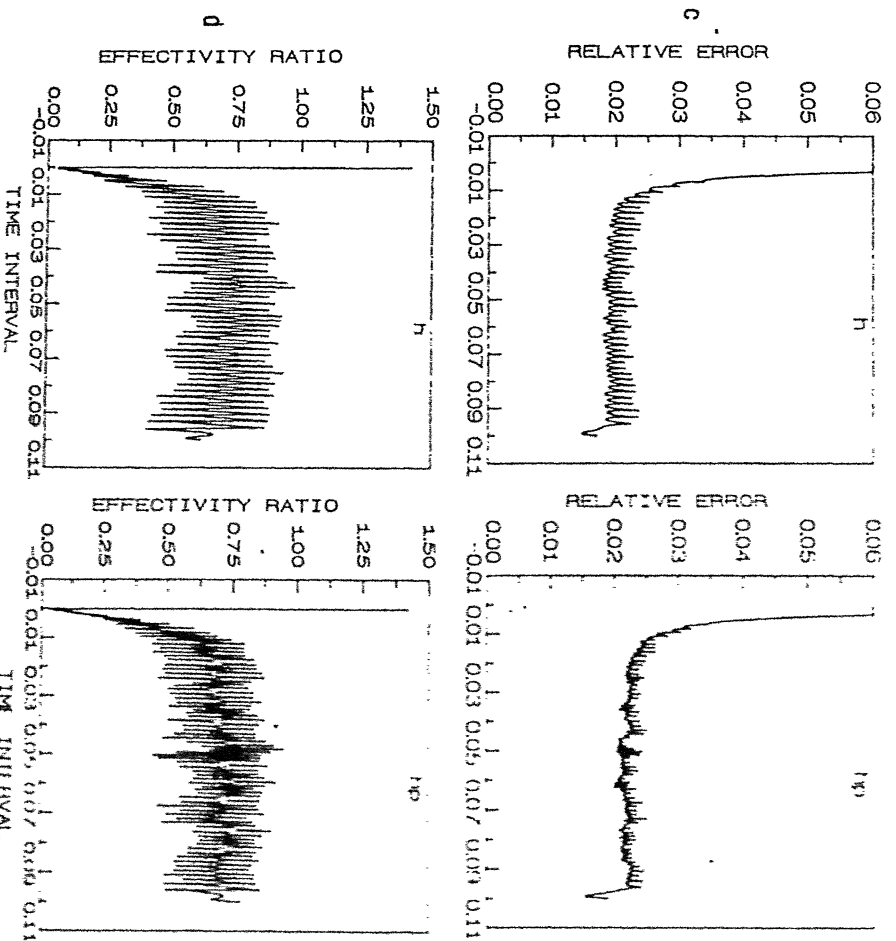
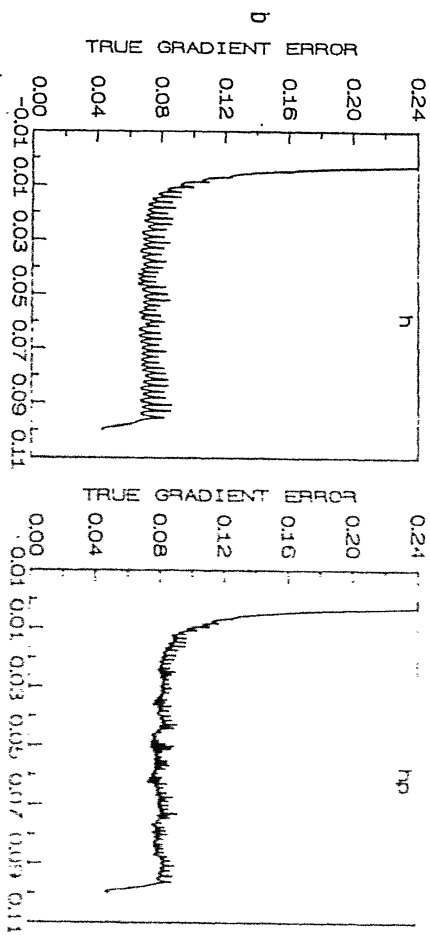
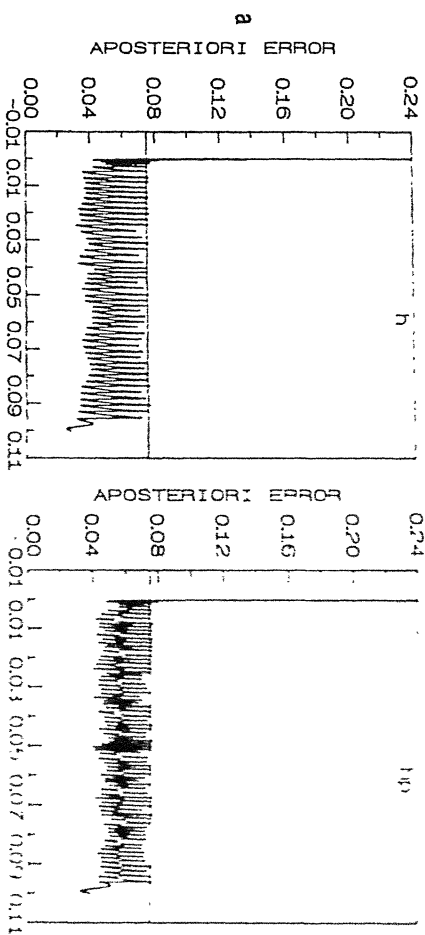


Fig. 412. TEST PROBLEM : TIME EVOLUTION OF ERRORS (h and hp version)



towards zero for initial times and then increases slowly to reach a mean value around 0.7. Initially, the *a posteriori* error which is based on the residue is very large for the coarse mesh, while the true gradient error is small due to the small values of computed and exact gradients. Therefore, the effectivity index is large. However, during the start-up period, the gradient at the wall increases (thereby resulting in larger true gradient error), while the residue is brought down rapidly by grid refinement. Due to this, the effectivity index reduces almost to zero. In the front propagation stage, both the *a posteriori* and true gradient errors are of similar magnitude. So, the effectivity index increases. It is apparent from the Figs. 4.12a and 4.12b that the *a posteriori* error is more sensitive to grid refinement than the true gradient error. Thus, as the front moves, adjustment of the grid from time to time brings down the *a posteriori* error considerably but not the true gradient error. So, the effectivity index does not equal unity for non-uniform meshes.

(C). The maximum and mean errors :

In Figs. 4.13a and 4.13b, the variations of the maximum and the mean elemental error indicators with respect to time have been presented for the *h* and *hp* version methods. In the initial period, the mean errors are very high for both the refinement strategies, due to the coarse starting- mesh. Following the application of the statistical procedure for grid refinement, many elements are stuffed in the large error zone and this leads to a rapid decrease in the values of the maximum as well as the mean errors. Indeed, since the statistical procedure is applied on

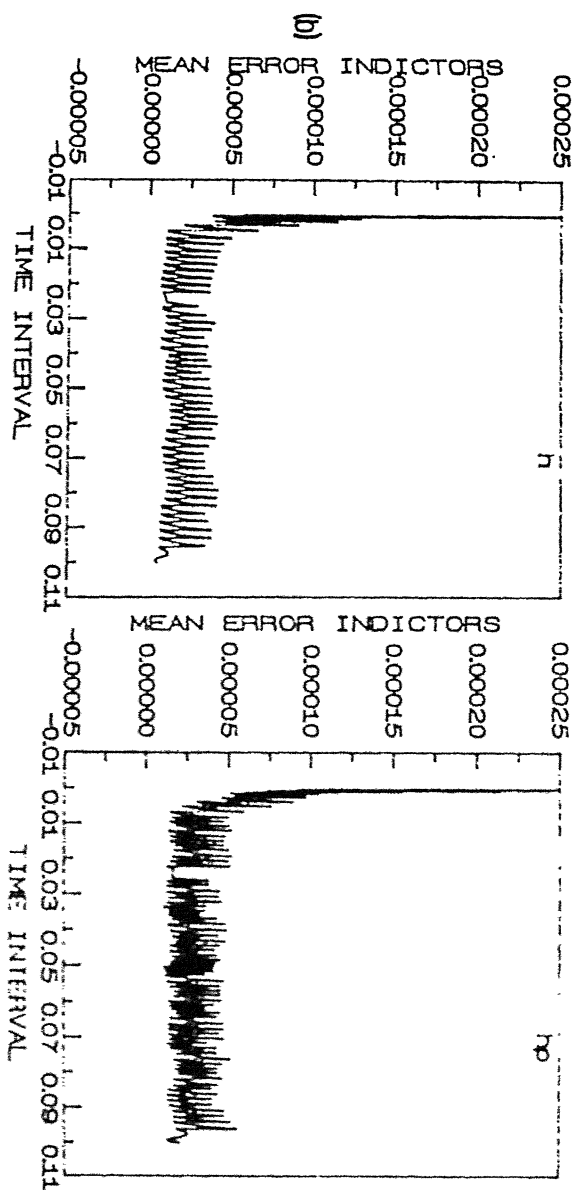
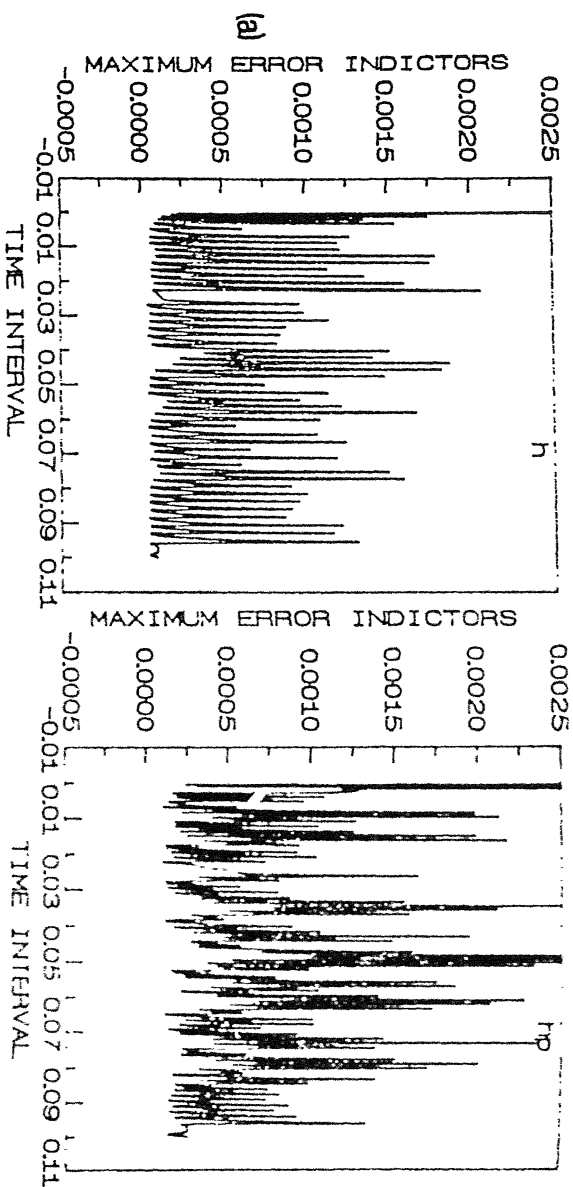


Fig. 4.13. TEST PROBLEM : TIME EVOLUTION OF ERRORS (h and h_p version)
(MAXIMUM AND MEAN ERROR INDICATORS)



successive meshes several times until the error tolerance criteria are met, within one time step the maximum and mean errors are reduced by a few orders of magnitude. Moreover, a natural gradation of grid spacing results which is well in accordance with the solution activity and consequently the local residue distribution approaches zero. In the moving front region, the mean error undergoes mild fluctuations due to the fact that there is some time lag between front movement and grid adjustment; in other words, the grid-adaption occurs only after sufficient residue has accumulated. This inference is supported by the trend that the maximum error increases with time sharply, reaches a peak value and then decreases rapidly due to grid-adaption. On the converged mesh, the maximum error is close to the mean, indicating an effective equi- distribution of error by the statistical technique (Fig. 4.13a).

(d). Adaptive mesh :

Figs. 4.14a and 4.14b show the adapted meshes at various time instants in the h and hp refinement procedures. The final adaptive meshes have been plotted after the convergence criterion is satisfied. Also, the number of elements have been indicated in the side tables presented along with the figures. The computations have been started with a uniform mesh of $N = 10$ elements. Further, the results of the h and h - p algorithms have also been presented in the tables (4.4a,b) at different time points.

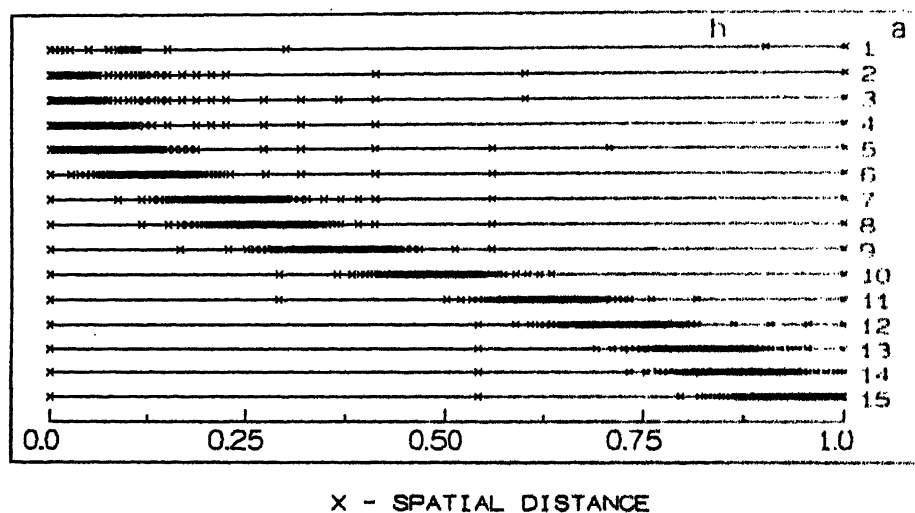
The figures clearly illustrate the ability of the adaptive scheme to concentrate grid points around the characteristic wave

| Time | Adp | Elements | | | N | Nodes | Apos.err | Grd.err | Eff.inc |
|---------|-----|----------|-----|-----|-----|-------|----------|---------|---------|
| | | Rem | Ref | Stf | | | | | |
| 0.00005 | 1 | 0 | 0 | 0 | 10 | 11 | 2.9327 | 2.0663 | 1.4193 |
| | 2 | 2 | 1 | 4 | 11 | 12 | 0.6807 | 1.8649 | 0.3650 |
| | 3 | 0 | 1 | 4 | 14 | 15 | 0.3248 | 1.8500 | 0.1756 |
| | 4 | 3 | 3 | 8 | 16 | 17 | 0.1357 | 1.8266 | 0.0743 |
| | 5 | 3 | 2 | 8 | 19 | 20 | 0.0557 | 1.8297 | 0.0305 |
| 0.00010 | 1 | 0 | 0 | 0 | 19 | 20 | 0.1374 | 1.6721 | 0.0822 |
| | 2 | 0 | 4 | 10 | 25 | 26 | 0.0708 | 1.6706 | 0.0424 |
| 0.00045 | 1 | 0 | 0 | 0 | 34 | 35 | 0.0828 | 1.0297 | 0.0804 |
| | 2 | 0 | 9 | 22 | 47 | 48 | 0.0422 | 1.0288 | 0.0411 |
| 0.00110 | 1 | 0 | 0 | 0 | 47 | 48 | 0.0777 | 0.5732 | 0.1356 |
| | 2 | 5 | 8 | 18 | 52 | 53 | 0.0476 | 0.5718 | 0.0833 |
| 0.00190 | 1 | 0 | 0 | 0 | 52 | 53 | 0.0772 | 0.3600 | 0.2143 |
| | 2 | 3 | 9 | 24 | 64 | 65 | 0.0505 | 0.3586 | 0.1409 |
| 0.00305 | 1 | 0 | 0 | 0 | 64 | 65 | 0.0757 | 0.2334 | 0.3245 |
| | 2 | 3 | 22 | 48 | 87 | 88 | 0.0424 | 0.2313 | 0.1834 |
| 0.00310 | 1 | 0 | 0 | 0 | 87 | 88 | 0.0424 | 0.2271 | 0.1868 |
| 0.00875 | 1 | 0 | 0 | 0 | 133 | 134 | 0.0762 | 0.1094 | 0.6960 |
| | 2 | 22 | 19 | 44 | 136 | 137 | 0.0394 | 0.1019 | 0.3866 |
| 0.00880 | 1 | 0 | 0 | 0 | 136 | 137 | 0.0377 | 0.0999 | 0.3774 |
| 0.01255 | 1 | 0 | 0 | 0 | 142 | 143 | 0.0753 | 0.0921 | 0.8172 |
| | 2 | 31 | 26 | 58 | 143 | 144 | 0.0403 | 0.0844 | 0.4781 |
| 0.02040 | 1 | 0 | 0 | 0 | 160 | 161 | 0.0727 | 0.0819 | 0.8871 |
| 0.02050 | 1 | 0 | 0 | 0 | 160 | 161 | 0.0764 | 0.0838 | 0.9110 |
| | 2 | 37 | 29 | 66 | 160 | 161 | 0.0351 | 0.0769 | 0.4570 |
| 0.02055 | 1 | 0 | 0 | 0 | 160 | 161 | 0.0338 | 0.0754 | 0.4480 |
| 0.03155 | 1 | 0 | 0 | 0 | 146 | 147 | 0.0396 | 0.0710 | 0.5570 |
| 0.05005 | 1 | 0 | 0 | 0 | 153 | 154 | 0.0757 | 0.0839 | 0.9024 |
| | 2 | 39 | 24 | 58 | 148 | 149 | 0.0373 | 0.0773 | 0.4832 |
| 0.05030 | 1 | 0 | 0 | 0 | 148 | 149 | 0.0370 | 0.0721 | 0.5134 |
| 0.06005 | 1 | 0 | 0 | 0 | 127 | 128 | 0.0725 | 0.0820 | 0.8846 |
| 0.06930 | 1 | 0 | 0 | 0 | 149 | 150 | 0.0364 | 0.0700 | 0.5204 |
| 0.07880 | 1 | 0 | 0 | 0 | 132 | 133 | 0.0613 | 0.0744 | 0.8240 |
| 0.08380 | 1 | 0 | 0 | 0 | 140 | 141 | 0.0761 | 0.0847 | 0.8984 |
| | 2 | 29 | 24 | 60 | 147 | 148 | 0.0342 | 0.0769 | 0.4447 |
| 0.08405 | 1 | 0 | 0 | 0 | 147 | 148 | 0.0347 | 0.0701 | 0.4942 |
| 0.09080 | 1 | 0 | 0 | 0 | 153 | 154 | 0.0766 | 0.0867 | 0.8833 |
| | 2 | 36 | 24 | 58 | 151 | 152 | 0.0333 | 0.0784 | 0.4245 |
| 0.09105 | 1 | 0 | 0 | 0 | 151 | 152 | 0.0349 | 0.0710 | 0.4916 |
| 0.09305 | 1 | 0 | 0 | 0 | 151 | 152 | 0.0755 | 0.0862 | 0.8753 |
| | 2 | 36 | 24 | 58 | 149 | 150 | 0.0322 | 0.0783 | 0.4110 |
| 0.09880 | 1 | 0 | 0 | 0 | 139 | 140 | 0.0298 | 0.0496 | 0.6010 |
| 0.09955 | 1 | 0 | 0 | 0 | 139 | 140 | 0.0248 | 0.0440 | 0.5638 |

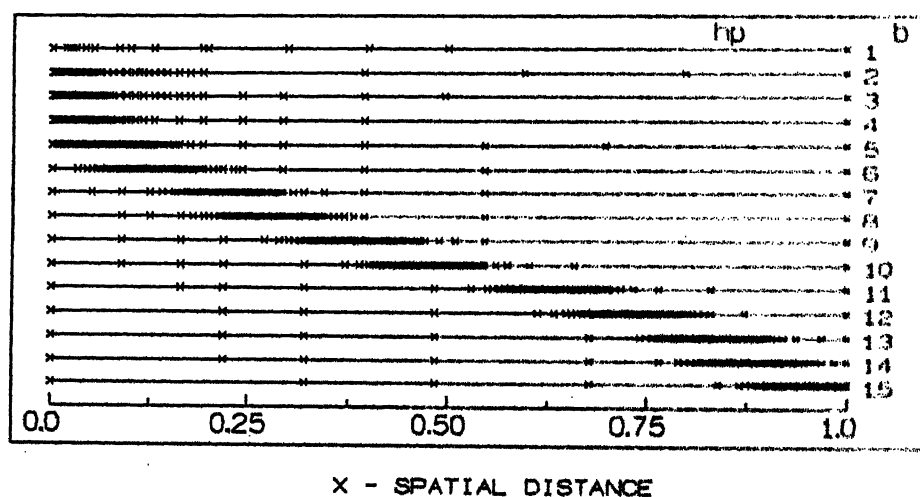
Table 4.4a. Test problem results for h-version strategy

| Time | Adp | Deg | Rem | N | Dof | Apos. err | Grd.err | Eff.ind |
|---------|-----|-----|-----|-----|-----|-----------|---------|---------|
| 0.00005 | 1 | 1 | 0 | 10 | 11 | 2.9327 | 2.0663 | 1.4193 |
| | 2 | 1 | 2 | 8 | 9 | 2.9327 | 2.0663 | 1.4193 |
| | 3 | 2 | 0 | 8 | 10 | 1.3210 | 2.0599 | 0.6413 |
| | 4 | 3 | 0 | 8 | 12 | 1.0376 | 2.0658 | 0.5022 |
| | 5 | 4 | 0 | 8 | 14 | 0.5135 | 2.0620 | 0.2490 |
| | 6 | 5 | 0 | 8 | 15 | 0.5498 | 2.0620 | 0.2666 |
| | 7 | 1 | 0 | 13 | 14 | 0.4364 | 1.8655 | 0.2339 |
| | 8 | 1 | 1 | 12 | 13 | 0.4364 | 1.8655 | 0.2339 |
| | 9 | 1 | 1 | 11 | 12 | 0.4364 | 1.8655 | 0.2339 |
| | 10 | 2 | 0 | 11 | 14 | 0.1708 | 1.8913 | 0.0903 |
| | 11 | 3 | 0 | 11 | 17 | 0.1900 | 1.8398 | 0.1033 |
| | 12 | 4 | 0 | 11 | 19 | 0.3038 | 1.7638 | 0.1723 |
| | 13 | 5 | 0 | 11 | 20 | 0.1759 | 1.7897 | 0.0983 |
| | 14 | 1 | 0 | 15 | 16 | 0.2903 | 1.8350 | 0.1582 |
| | 15 | 2 | 0 | 15 | 20 | 0.1257 | 1.8272 | 0.0688 |
| | 16 | 3 | 0 | 15 | 23 | 0.0630 | 1.8186 | 0.0347 |
| 0.00010 | 1 | 1 | 0 | 22 | 23 | 0.0890 | 1.6685 | 0.0533 |
| | 2 | 1 | 2 | 20 | 21 | 0.0890 | 1.6685 | 0.0533 |
| | 3 | 1 | 1 | 19 | 20 | 0.0890 | 1.6685 | 0.0533 |
| | 4 | 2 | 0 | 19 | 25 | 0.0552 | 1.6696 | 0.0330 |
| 0.00030 | 1 | 1 | 0 | 30 | 31 | 0.0568 | 1.2368 | 0.0459 |
| 0.00055 | 1 | 1 | 0 | 30 | 31 | 0.0801 | 0.9232 | 0.0867 |
| | 2 | 2 | 0 | 30 | 39 | 0.0486 | 0.9279 | 0.0524 |
| 0.00665 | 1 | 1 | 0 | 95 | 96 | 0.0751 | 0.1252 | 0.5998 |
| | 2 | 2 | 0 | 95 | 122 | 0.0419 | 0.1262 | 0.3318 |
| 0.01750 | 1 | 1 | 0 | 118 | 119 | 0.0763 | 0.0916 | 0.8325 |
| | 2 | 1 | 1 | 117 | 118 | 0.0763 | 0.0916 | 0.8325 |
| | 3 | 2 | 0 | 117 | 130 | 0.0461 | 0.0919 | 0.5012 |
| 0.03080 | 1 | 1 | 0 | 146 | 147 | 0.0446 | 0.0840 | 0.5313 |
| 0.03830 | 1 | 1 | 0 | 119 | 120 | 0.0751 | 0.0859 | 0.8738 |
| | 2 | 1 | 1 | 118 | 119 | 0.0751 | 0.0857 | 0.8762 |
| | 3 | 2 | 0 | 118 | 146 | 0.0432 | 0.0857 | 0.5042 |
| 0.03855 | 1 | 1 | 0 | 142 | 143 | 0.0442 | 0.0802 | 0.5512 |
| 0.06980 | 1 | 1 | 0 | 130 | 131 | 0.0610 | 0.0800 | 0.7622 |
| 0.09080 | 1 | 1 | 0 | 122 | 123 | 0.0756 | 0.0901 | 0.8394 |
| | 2 | 1 | 1 | 121 | 122 | 0.0756 | 0.0900 | 0.8407 |
| | 3 | 2 | 0 | 121 | 137 | 0.0443 | 0.0900 | 0.4920 |
| 0.09105 | 1 | 1 | 0 | 134 | 135 | 0.0482 | 0.0829 | 0.5816 |
| | 2 | 1 | 1 | 133 | 134 | 0.0482 | 0.0829 | 0.5816 |
| 0.09955 | 1 | 1 | 0 | 81 | 82 | 0.0334 | 0.0472 | 0.7088 |
| | 2 | 1 | 1 | 80 | 81 | 0.0334 | 0.0472 | 0.7089 |
| 0.09980 | 1 | 1 | 0 | 78 | 79 | 0.0360 | 0.0480 | 0.7484 |
| 0.10000 | 1 | 1 | 0 | 75 | 76 | 0.0384 | 0.0492 | 0.7801 |

Table 4.4b. Test problem results for h-p version strategy



| Act. | Time | Eles | Nodes |
|------|----------|------|-------|
| 1 | 0.000050 | 19 | 20 |
| 2 | 0.001250 | 52 | 53 |
| 3 | 0.002500 | 64 | 65 |
| 4 | 0.005000 | 119 | 120 |
| 5 | 0.010000 | 136 | 137 |
| 6 | 0.015000 | 148 | 149 |
| 7 | 0.025000 | 159 | 160 |
| 8 | 0.030000 | 148 | 149 |
| 9 | 0.040000 | 157 | 158 |
| 10 | 0.050000 | 148 | 149 |
| 11 | 0.065000 | 140 | 141 |
| 12 | 0.075000 | 140 | 141 |
| 13 | 0.085000 | 147 | 148 |
| 14 | 0.090000 | 153 | 154 |
| 15 | 0.097800 | 139 | 140 |



| Act. | Time | Eles | Nodes |
|------|----------|------|-------|
| 1 | 0.000050 | 15 | 16 |
| 2 | 0.001250 | 55 | 56 |
| 3 | 0.002500 | 68 | 69 |
| 4 | 0.005000 | 76 | 77 |
| 5 | 0.010000 | 134 | 135 |
| 6 | 0.015000 | 139 | 140 |
| 7 | 0.025000 | 127 | 128 |
| 8 | 0.030000 | 127 | 128 |
| 9 | 0.040000 | 139 | 140 |
| 10 | 0.050000 | 142 | 143 |
| 11 | 0.065000 | 123 | 124 |
| 12 | 0.075000 | 121 | 122 |
| 13 | 0.085000 | 132 | 133 |
| 14 | 0.090000 | 130 | 131 |
| 15 | 0.097800 | 98 | 99 |

Fig.4.14 TEST PROBLEM (h & hp version) ADAPTED MESH AT VARIOUS t

$x - Ct = 0$, where all the activity takes place. In the h refinement procedure, the adaptive algorithm automatically removes, retains and stuffs nodes simultaneously. It is however, possible that some unnecessary nodes exist in regions where no solution activity is taking place. In the present work, removal of nodes has been carried out if the error of the neighbouring elements goes below a minimum error tolerance of 10^{-6} . Some times this tolerance value is very much below the mean error and so even for regions with no solution activity, node removal does not take place. In order to eliminate this problem, dynamic setting of the minimum tolerance limit (as a fraction of the mean error) can be done, as discussed in Chapter 5. On the other hand, over refinement of the grid occurs some times due to the statistical refinement procedure. In this scheme, if the local error is greater than the mean, element refinement is performed. At later stages of grid adaption, the mean error is very small and it is prone to be affected by round off error. Thus, at few instants, elements which need not be refined are also divided further. Occasionally, excessive refinement may lead to deterioration of the computed solution due to domination of round off errors; but this can be easily checked by specifying a minimum grid spacing. These difficulties are, however, rare exceptions and as seen from the Figs. 4.14a-b, the performance of the h and h - p refinement strategies is remarkably good. The higher degree approximations of the p version tend to produce oscillations in the solution. Therefore, the local elemental errors may not behave well, especially adjacent to the moving front. Due to these oscillations, some nodes are not properly removed from low

gradient regions during hp adaption. From both Figs. 4.14a-b, it is clear that the nodes are distributed in anticipation of the front motion.

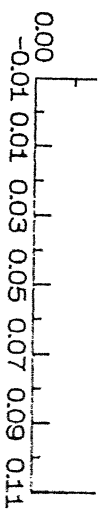
Comparing the relative performance between the h and hp refinement methods, the following observations can be made. In general, the h-version is more efficient in bringing down the error than the hp- version. Therefore, both the maximum and mean error values are lower in the h-version (Fig. 4.13a and 4.13b). Further, after each grid adaption, the maximum error is brought very close to the mean error by the h refinement strategy, implying a very effective equi-distribution of error. On a comparative scale, the degree of equi-distribution by the hp refinement method is not very good. Also, the error decreases slowly when the degree of interpolation is increased by the statistical procedure. In the present work, if the maximum allowed polynomial degree is reached without reaching error convergence, the hp -version mesh is converted into an approximately equivalent h-version mesh. After this conversion, the error may slightly increase or decrease. Due to the slow convergence, the hp method requires more number of adaptations. The higher values of the maximum and the mean errors observed in the hp version can be attributed to the following reasons. It is observed that for interpolation order greater than 3, the computed solution is more prone to numerical noise, which is seen in the form of oscillations in the computed solution. Also, as the degree of interpolation is increased in one element, the computed error of the neighbouring elements tends to increase. On various accounts, the performance of the h-version refinement appears to

be superior, although the number of elements required by the hp version is less in order to meet the same error criterion. The computational effort per adaption is also less in the h-p version due to the application of the hierarchical basis function concept.

(e). Effect of solution smoothening on true gradient error estimation :

The effects of the smoothening procedure upon the error measures are shown in the Figs. 4.15a-c. Due to the linear degree approximation used in the h-version procedure, the computed gradient is constant within each element. On the other hand, the exact gradient varies from point to point even within an element. For this reason, piece-wise cubic spline smoothening procedure has been applied to the computed solution in each element, for improving the estimates of true gradient error and the effectivity index. For smoothening purposes, the average value between the forward and backward derivatives of the gradient at the nodes are first evaluated. Using these average gradients as well as the values of the computed solutions at the nodes, piece-wise cubic Hermite interpolating polynomials have been employed for smoothening.

It is seen that the true gradient and relative errors are somewhat reduced after smoothening, thereby bringing the value of the effectivity index closer to unity for the non-uniform mesh. However, the magnitudes of the oscillations in the true gradient error and the effectivity index increase after smoothening which is not an acceptable feature. In any case, the previously



c

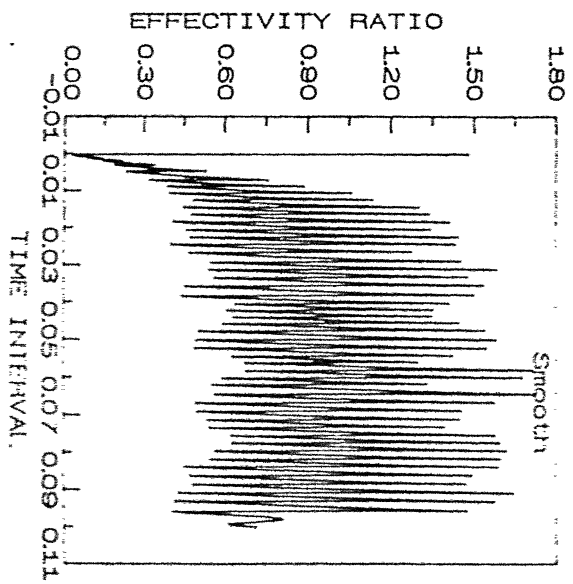
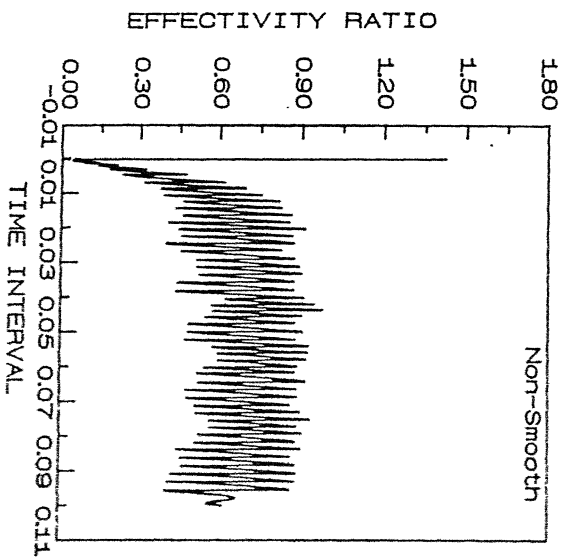
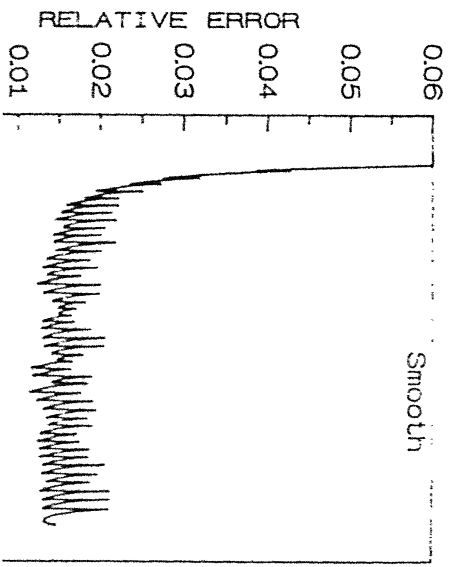
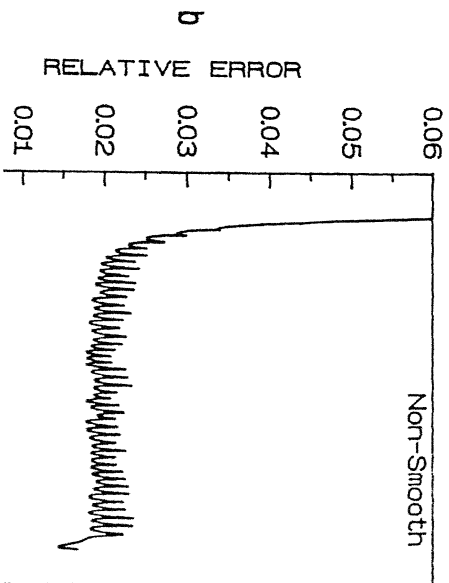
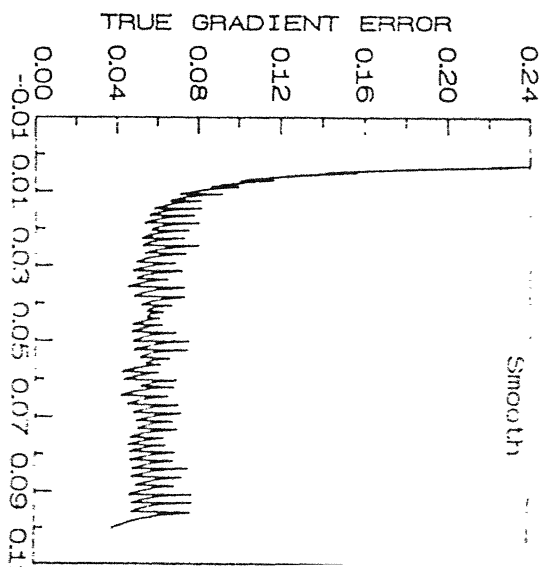
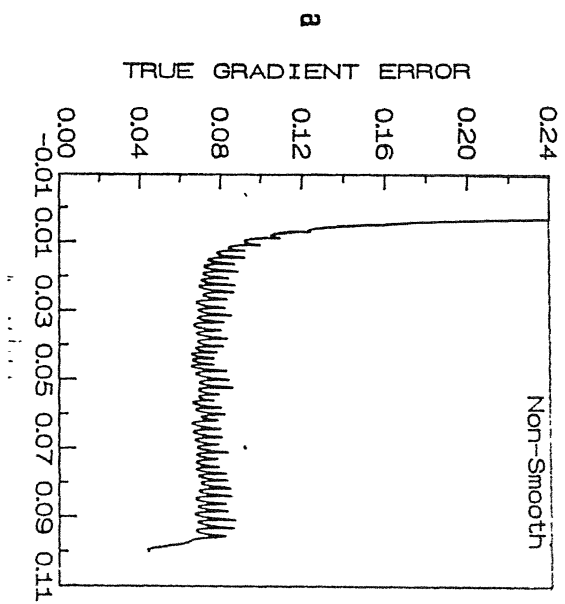


Fig. 415. TEST PROBLEM : TIME EVOLUTION OF ERRORS (h VERSION)
(NON-SMOOTH AND SMOOTH EFFECT)



observed trend that the *a posteriori* error is slightly less than the true gradient error for non-uniform meshes, holds with smoothening also. For the hp refinement method, due to its inherently smooth nature of solution (of higher degree approximating polynomials), the smoothening of the computed solutions has not been performed while estimating true gradient error for the test problem.

(III). Grid speed :

On a coarse mesh the grid speed calculation is inaccurate. So, we consider the solution at a particular time when adequate h-refinement has been achieved and a fully established front moves at a uniform velocity.

Fig. 4.16. shows the behaviour of solution profile at various times during the uniform front propagation stage. The exact solution and the solution by grid speed have been superimposed. It is observed from the figure that the computed solution has good correspondence with the exact solution. Because of the highly non-uniform mesh which has been considered here, the solution front moves in a smooth way. However, some difficulties do arise while predicting the solution in a small neighbourhood near the boundaries $x = 0.0$ and $x = 1.0$. The reason for this is that if all the grid points move with same velocity, collapsing and stretching of elements occur near the boundary which affect the solutions. In order to avoid this, special treatment of grid movement is necessary near the domain boundaries.

In the computational experiments, the boundary nodes of the domain have been locked and the interior nodes are allowed to move

according to uniform grid speed. After a few time iterations, the nodes which are near to the boundary may shoot outside the solution domain. These nodes which cross the boundary are reintroduced at the beginning of the mesh, as seen in the Fig. 4.17. Although, it is not necessary to reintroduce these nodes in the present problem, for more general cases it may be necessary. It is clear from the Fig. 4.17. that the nodes are distributed according to the front. The computed grid velocity for this problem matches very well with the exact front speed value of 10.

Fig. 4.18a. shows the variation of a *posteriori* error with time in the wave front region. It is observed that the *a posteriori* error increases very slowly, but for the entire wave propagation time, the error is very small (smaller than the tolerance used for h and h_p methods). This shows that the elemental error indicators also move according to front. The small magnitude of the errors also implies that the chosen initial mesh(obtained by h - version) is adequate for resolving the sharp spatial gradients. The slow increase in error with respect to time may be attributed to a small lag between the grid movement and the front movement.

A similar behaviour is observed in the Fig. 4.18b in which the true gradient error has been plotted. The decrease in the true gradient error at initial time shows that the gradient of the space discretization error decreases rapidly due to very fine mesh around the front location at that time. At later times, since the grid movement does not exactly coincide with the front, true gradient error increases slowly. The relative error behaviour

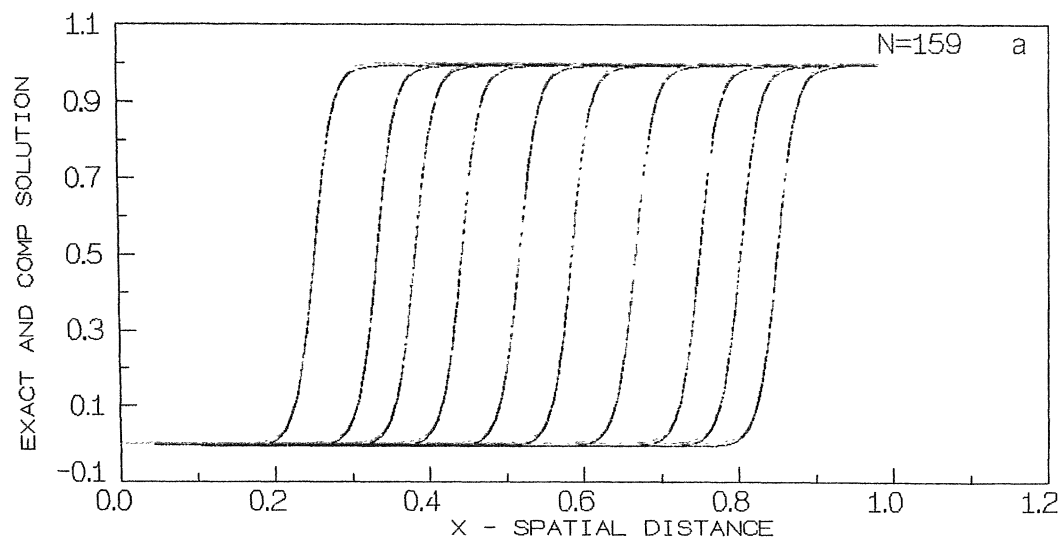


Fig. 4.16 TEST PROBLEM (Grid speed) : SOLUTION PROFILE

| Solution | Line_type |
|------------|-----------|
| Grid speed | ----- |
| Exact soln | - - - - - |

Fig. (a)

| Adp | Time | Eles | Nods |
|-----|-----------|------|------|
| 1 | 0 .025200 | 159 | 160 |
| 2 | 0 .033300 | 159 | 160 |
| 3 | 0 .038250 | 159 | 160 |
| 4 | 0 .044250 | 159 | 160 |
| 5 | 0 .051750 | 159 | 160 |
| 6 | 0 .058500 | 159 | 160 |
| 7 | 0 .066750 | 159 | 160 |
| 8 | 0 .075000 | 159 | 160 |
| 9 | 0 .080250 | 159 | 160 |
| 10 | 0 .085000 | 159 | 160 |

Fig. (b)

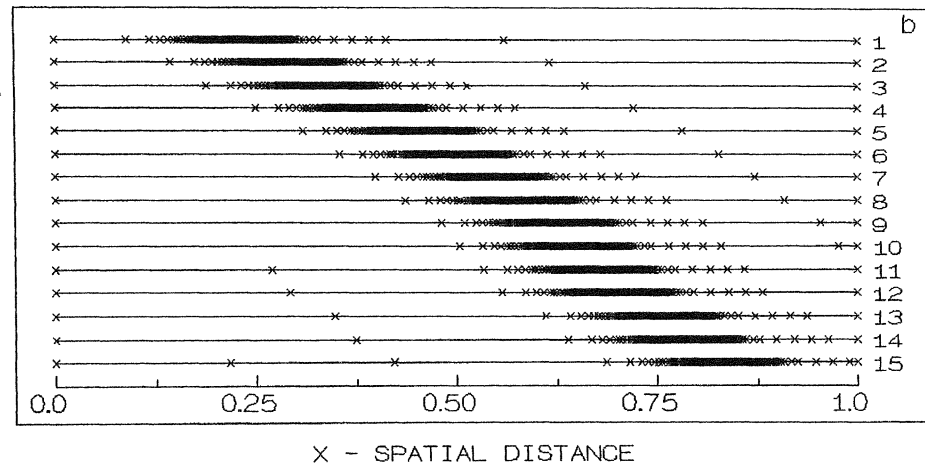


Fig. 4.17 TEST PROBLEM (Grid speed) : ADAPTED MESH AT VARIOUS t

| Adp | Time | Eles | Nods |
|-----|-----------|------|------|
| 1 | 0 .025200 | 159 | 160 |
| 2 | 0 .030750 | 159 | 160 |
| 3 | 0 .035250 | 159 | 160 |
| 4 | 0 .041250 | 159 | 160 |
| 5 | 0 .047250 | 159 | 160 |
| 6 | 0 .051750 | 159 | 160 |
| 7 | 0 .056250 | 159 | 160 |
| 8 | 0 .060000 | 159 | 160 |
| 9 | 0 .064500 | 159 | 160 |
| 10 | 0 .066750 | 159 | 160 |
| 11 | 0 .069750 | 159 | 160 |
| 12 | 0 .072000 | 159 | 160 |
| 13 | 0 .077550 | 159 | 160 |
| 14 | 0 .080250 | 159 | 160 |
| 15 | 0 .085000 | 159 | 160 |

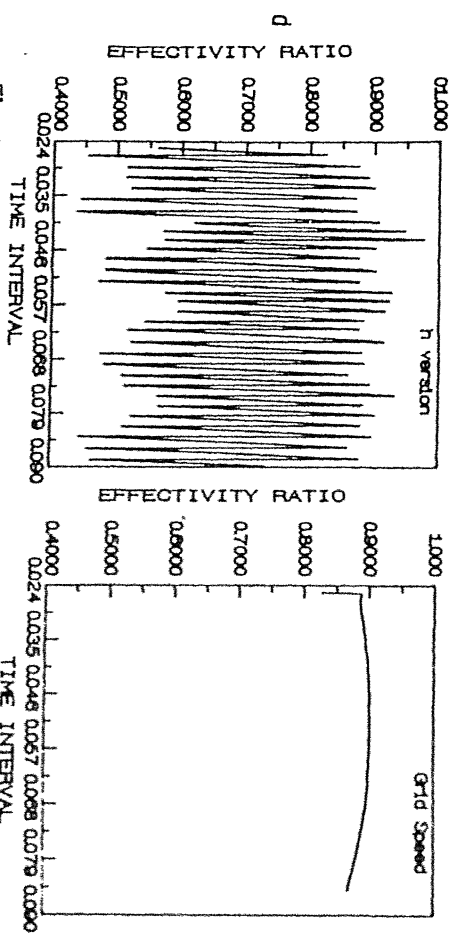
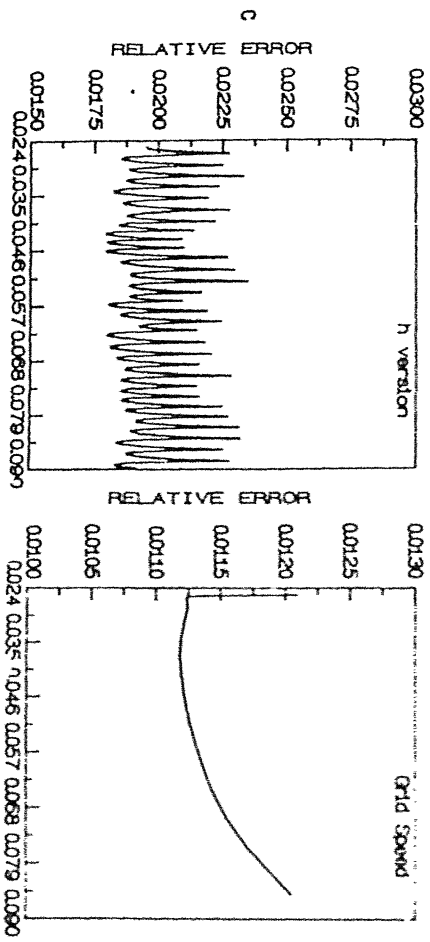
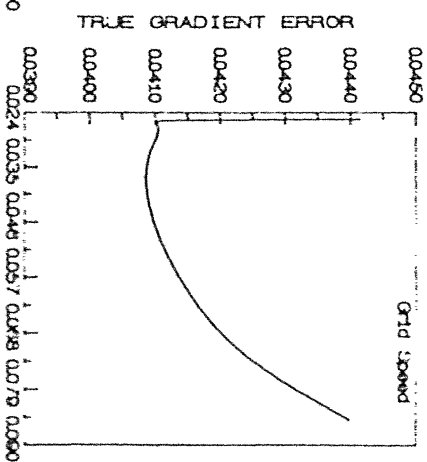
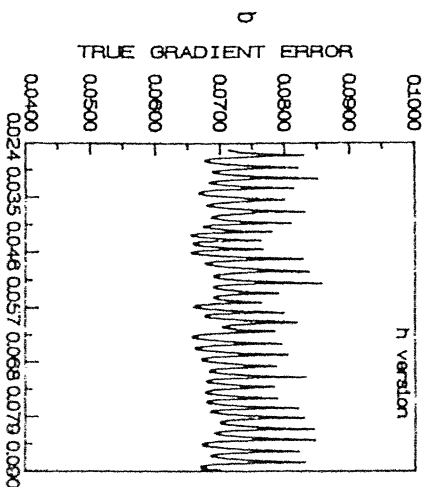
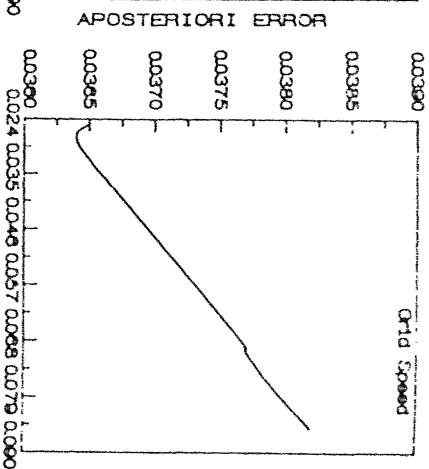
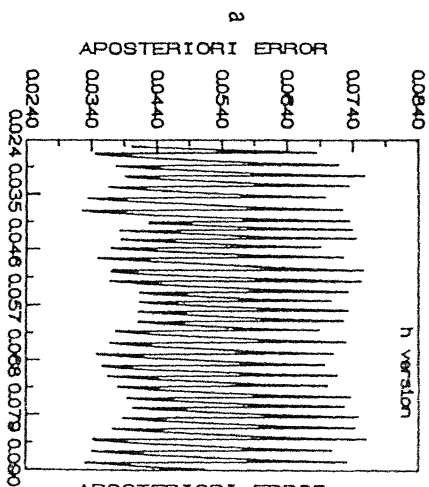


Fig. 418. TEST PROBLEM : TIME EVOLUTION OF ERRORS (GRID SPEED)



shown in Fig. 4.18c is same as that of the gradient error.

Results for the effectivity index versus time are shown in Fig. 4.18d. From the graph, it can be seen that the effectivity index increases rapidly at initial times and it remains almost constant till the final time is reached. Its value is not too far from unity, indicating a proper relationship between the *a posteriori* and true gradient error estimates. From Figs. 4.18a-d, it is clear that the initial variations of these error measures for grid-speed method depend on the starting solution and the mesh. For later times, the variations are governed by the difference between the grid and the front speeds. If the starting solution is good and the estimate of the grid speed is accurate, error measures can be maintained below the tolerance level for the whole duration of the problem. It is, however, advisable to check whether the solution obtained from grid speed procedure satisfies error criteria or not, from time to time. Thus, the grid speed technique can be an inexpensive way of constructing adaptive meshes for problems with constant front speeds.

4.8.2. Application of adaptive strategy to the flame propagation problem :

In this section, results obtained by h-version, hp-version and grid-speed methods for the flame propagation problem described by equations (4.4) are presented. The solution profiles (temperature and reactant density) and the variation of error measures have been plotted at various time instants, for both uniform and non-uniform (adaptive) meshes. The evolution of adaptive meshes with time have also been shown for h, h-p and the

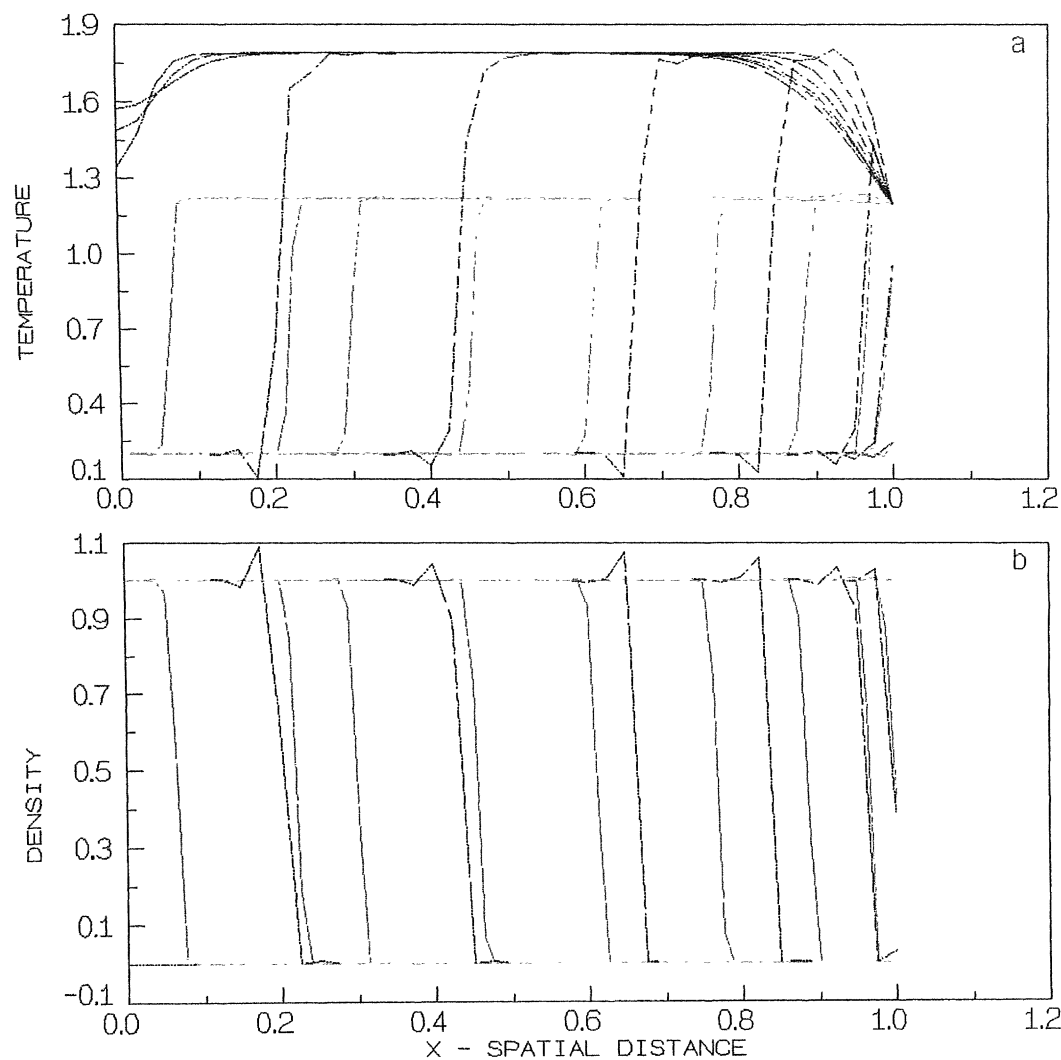
grid speed strategies.

(I). Uniform mesh results :

For the non -linear flame problem considered in the present study, exact solutions are not available, Therefore, it is not possible to estimate the true gradient error in the computed solution. However, in the previous section, the behaviour of the true gradient and the *a posteriori* errors were shown to be identical for the test problem which has similar characteristics as the flame problem. Therefore, a *a posteriori* error which is computable from the elemental residues, is taken as the error measure for grid- adaptations. Bieterman and Babuska (1982a-b; 1986) have also used the same error estimate for the problems of this nature. The flame problem was solved on a sequence of uniform meshes with 40, 80, 160 and 320 elements. The reasons for considering the uniform meshes have been cited in the previous section. For the flame problem also, uniform grid results can be used as reference, since they are free of the moving grid effects.

(a). Temperature and density profiles :

In Figs. 4.19a-b and 4.20a-b, the temperature and density profiles for the flame propagation problem have been shown for uniform meshes with 40, 80, 160 and 320 elements. For the coarse mesh ($N = 40$), the maximum flame temperature is unrealistically high; besides, oscillations are observed at the edges of the flame front due to inaccurate resolution of the gradients. In fact, as described in detail by Dwyer (1983b), the flame profile reaches its equilibrium shape some time after ignition, due to the balance

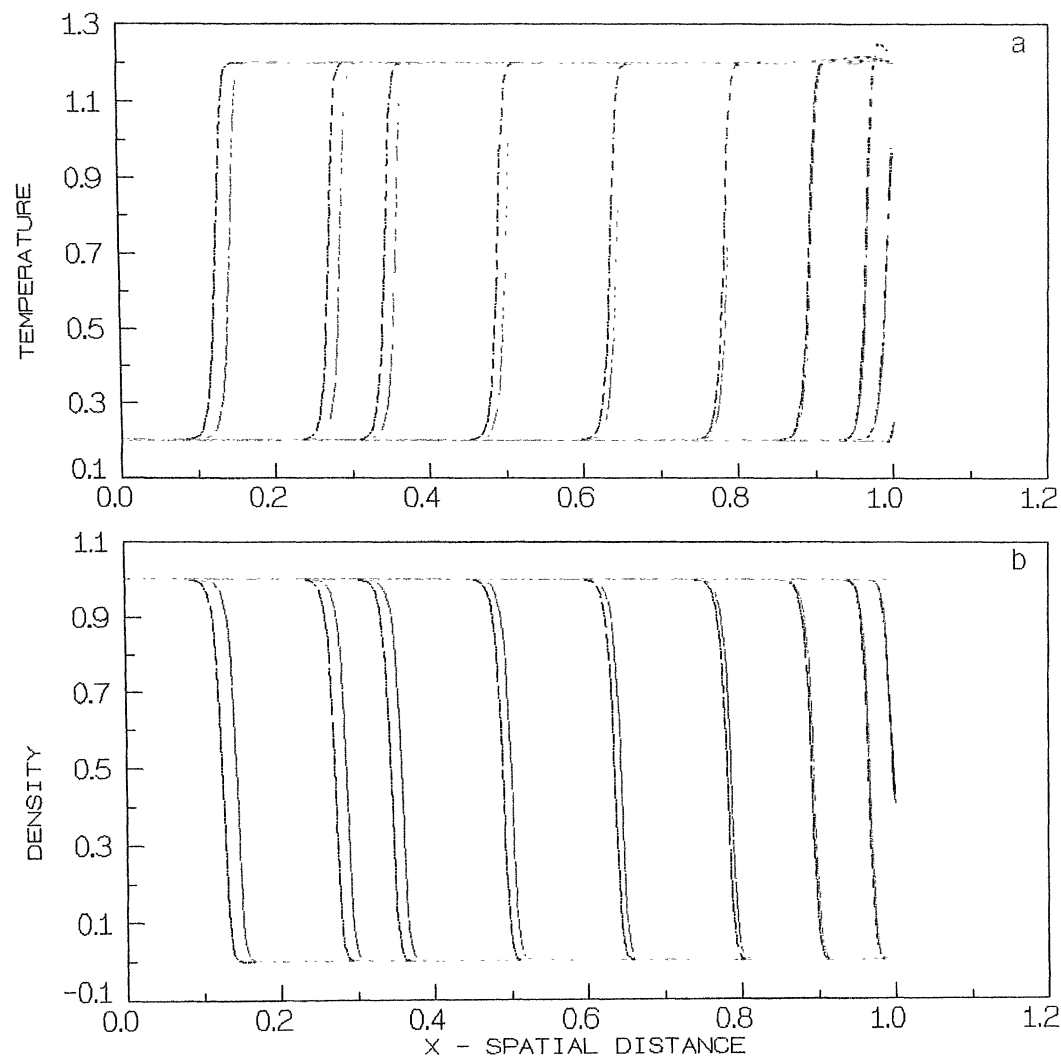


| Mesh | Line_type |
|--------|-----------|
| N = 40 | ----- |
| N = 80 | ----- |

| Adp | Time | Eles | Nods |
|-----|-----------|------|------|
| 1 | 0 .000010 | 40 | 41 |
| 2 | 0 .000150 | 40 | 41 |
| 3 | 0 .000300 | 40 | 41 |
| 4 | 0 .000800 | 40 | 41 |
| 5 | 0 .001550 | 40 | 41 |
| 6 | 0 .002550 | 40 | 41 |
| 7 | 0 .003550 | 40 | 41 |
| 8 | 0 .004550 | 40 | 41 |
| 9 | 0 .005050 | 40 | 41 |
| 10 | 0 .006050 | 40 | 41 |

| Adp | Time | Eles | Nods |
|-----|-----------|------|------|
| 1 | 0 .000010 | 80 | 81 |
| 2 | 0 .000150 | 80 | 81 |
| 3 | 0 .000300 | 80 | 81 |
| 4 | 0 .000800 | 80 | 81 |
| 5 | 0 .001550 | 80 | 81 |
| 6 | 0 .002550 | 80 | 81 |
| 7 | 0 .003550 | 80 | 81 |
| 8 | 0 .004550 | 80 | 81 |
| 9 | 0 .005050 | 80 | 81 |
| 10 | 0 .006050 | 80 | 81 |

Fig. 4.19 FLAME PROBLEM (Uniform Mesh) : SOLUTION PROFILE



| Mesh | Line_type |
|-----------|-------------|
| $N = 160$ | ----- |
| $N = 320$ | - . - . - . |

| Adp | Time | Eles | Nods |
|-----|-----------|------|------|
| 1 | 0 .000010 | 160 | 161 |
| 2 | 0 .000150 | 160 | 161 |
| 3 | 0 .000300 | 160 | 161 |
| 4 | 0 .000800 | 160 | 161 |
| 5 | 0 .001550 | 160 | 161 |
| 6 | 0 .002550 | 160 | 161 |
| 7 | 0 .003550 | 160 | 161 |
| 8 | 0 .004550 | 160 | 161 |
| 9 | 0 .005050 | 160 | 161 |
| 10 | 0 .006050 | 160 | 161 |

| Adp | Time | Eles | Nods |
|-----|-----------|------|------|
| 1 | 0 .000010 | 320 | 321 |
| 2 | 0 .000150 | 320 | 321 |
| 3 | 0 .000300 | 320 | 321 |
| 4 | 0 .000800 | 320 | 321 |
| 5 | 0 .001550 | 320 | 321 |
| 6 | 0 .002550 | 320 | 321 |
| 7 | 0 .003550 | 320 | 321 |
| 8 | 0 .004550 | 320 | 321 |
| 9 | 0 .005050 | 320 | 321 |
| 10 | 0 .006050 | 320 | 321 |

Fig. 4.20 FLAME PROBLEM (Uniform Mesh) : SOLUTION PROFILE

between the reaction and the heat conduction process. Since, the heat conduction process depends on the solution gradient, it is understandable that an inaccurate estimation of the temperature/density gradients will lead to an inaccurate prediction of the reaction rate and the flame temperature. The overshoot in flame temperature is eliminated for finer meshes and the oscillations in the solution profiles are also absent which can be seen from the Figs. 4.20a and 4.20b. For the coarse mesh, although the solutions are highly inaccurate, all the flame features such as equilibrium flame shape, constant speed of front movement etc., are observed. For more refined meshes (with $N = 80, 160$ or 320), converged equilibrium flame shapes are obtained. Just after ignition, the flame profile exhibits a small hump in its shape. The presence of the hump is attributed to the shift in equilibrium shape due to the higher heat conduction effect in the proximity of the wall ($x = 1.0$). When the flame moves away from the wall, the hump in the flame profile disappears.

Regarding the density profiles, the following observations are made. For coarse mesh ($N = 40$), the density profile also has fluctuations in the form of overshoots or undershoots near the edges of the flame. Indeed, after the flame front has moved through a particular location, the density of the reactants at that location is expected to go zero due to complete burning. However, for the coarse mesh, there is a tendency to predict small negative values of density in the burn-out region, due to numerical oscillations. In the computational scheme, these non physical negative values have been forcefully made equal to zero.

The reaction term acts as a heat source in the temperature equation, while it is a sink term for the reactant density equation. For this reason, the density profile has just the inverse shape of the temperature profile. All the other features such as front movement etc., are depicted by the density profile also. For the number of elements $N = 160$ and 320 , identical profiles are obtained, for each instant of time.

In general, for $N = 40$, the convergence of the temperature or density profiles posed some difficulties during iterative method of solution due to large fluctuations in the predicted values. For fine meshes, this convergence problem did not arise. The computed shapes and front locations widely differ between the meshes of $N = 40$ and 80 while excellent agreement is observed between the predictions of $N = 160$ and 320 . It is noted that, the front speed predicted by the coarse mesh is also highly inaccurate.

(b). The behaviour of a posteriori error estimate :

In Figs. 4.21a-d, the variation of the a posteriori error and the solution gradient with time are shown for different number of elements. The trends of a posteriori error seen for the flame problem are very similar to those of the test problem. During the ignition/extinction phases of the flame, the a posteriori error is small due to the low value of solution gradients. An interesting feature seen for the coarse ($N = 40$) mesh is that the flame extinguishes earlier compared to the finer meshes, due to higher flame velocity. In fact, computations have not been carried till extinction for $N = 80, 160$ and 320 .

In the flame propagation region systematic high amplitude

fluctuations are observed in the error, similar to the results of the test problem with uniform elements. For $N = 40$, the *a posteriori* error exhibits large oscillations and the amplitude of the fluctuations are very high. A similar trend has been observed for $N = 80$, although the magnitude of the fluctuations have been reduced.

As the number of elements increases, the amplitude of the fluctuations decreases and for $N > 160$, it is negligibly small. It is observed that for each mesh the maximum error occurs slightly after ignition when the hump formation is seen in the solution profile. This is due to the fact that the nonlinear combustion reaction is most active at that particular time, possibly because of high temperature occurring in the reactant mixture. When the flame front moves with a constant velocity, the process of combustion reaction, heat transfer and front movement are in perfect equilibrium with each other; thus the mean value of the *a posteriori* error reaches a steady value in this region. However, the magnitude of the this steady value of error, decreases with mesh size. Analogous to the results of the test problem, the error is seen to be linearly proportional to the mesh size, especially, for the fine meshes. For coarse meshes, the error seems to decrease faster than h , when the grid is refined. It is to be noted from the Figs. 4.21a-d that even on the fine mesh ($N = 320$) the *a posteriori* error does not meet the error tolerance criterion.

Thus, from the above discussions, it is clear that the error behaviour for the flame propagation problem has very similar characteristics as that of the test problem. Whatever differences

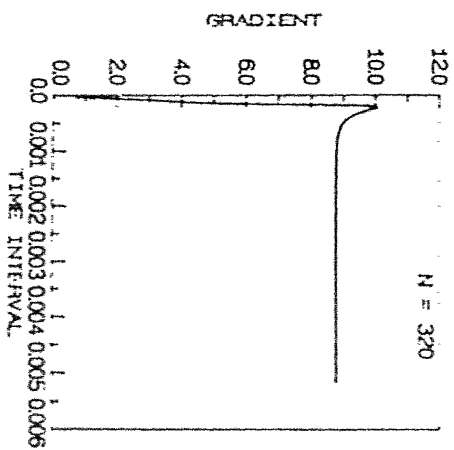
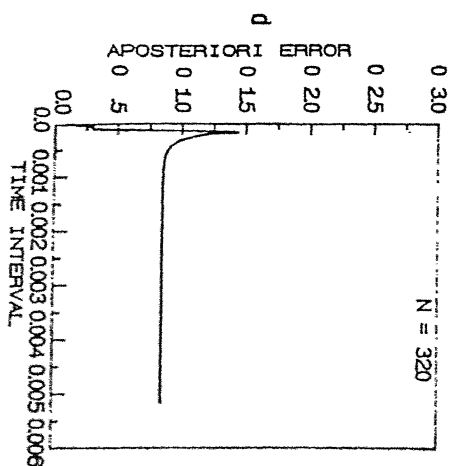
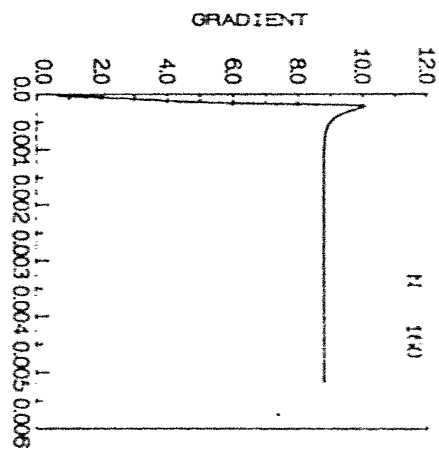
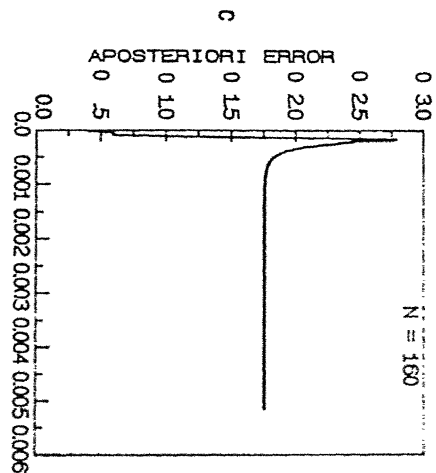
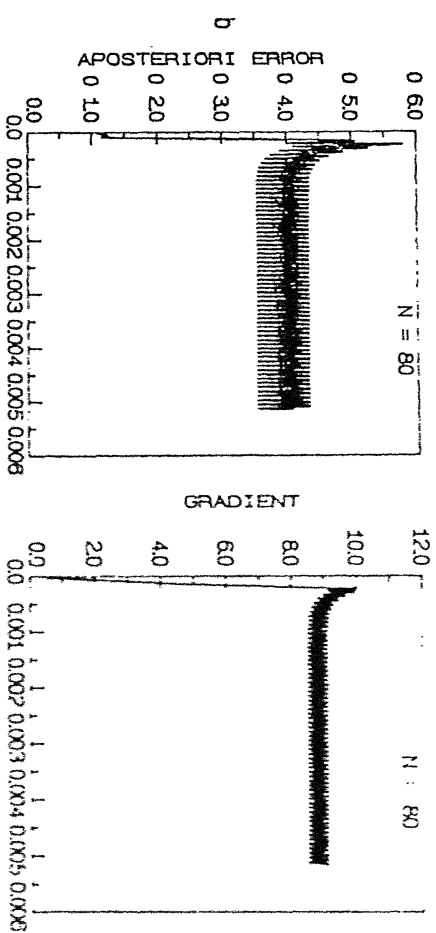
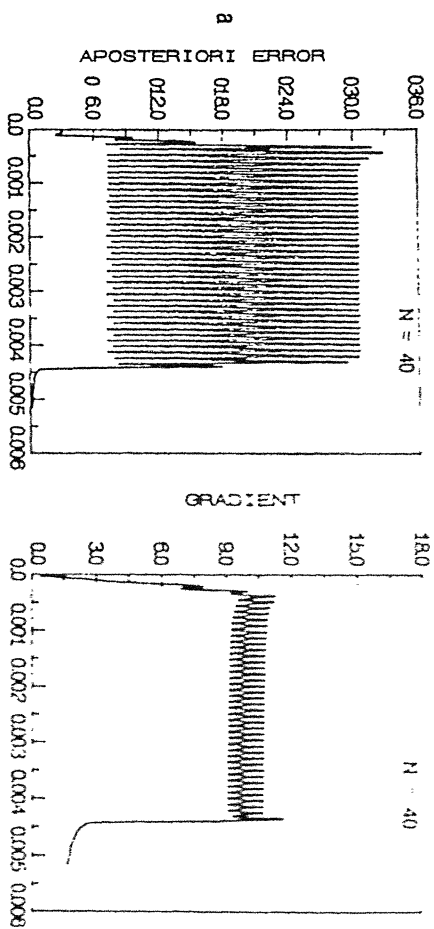


Fig. 4.21 FLAME PROBLEM : TIME EVOLUTION OF ERRORS (UNIFORM MESH)
(APOSTERIORI ERROR AND GRADIENT)



exist between the trends of these two problems, can be attributed to the inherent nonlinearity of the flame problem and also the differences in the front width as well as the front speed between the two problems.

(II). Non-uniform mesh results :

The analysis carried out by the uniform grids shows that a large number of grid points are necessary to resolve the flame front which satisfies the error tolerance criterion. So, the solution procedure becomes computationally very expensive when uniform grids are used. In order to have an accurate physical simulation, adaptive grids are necessary which automatically identify the location of the active zone. Results obtained with h , hp and grid speed strategies are presented in the ensuing sections.

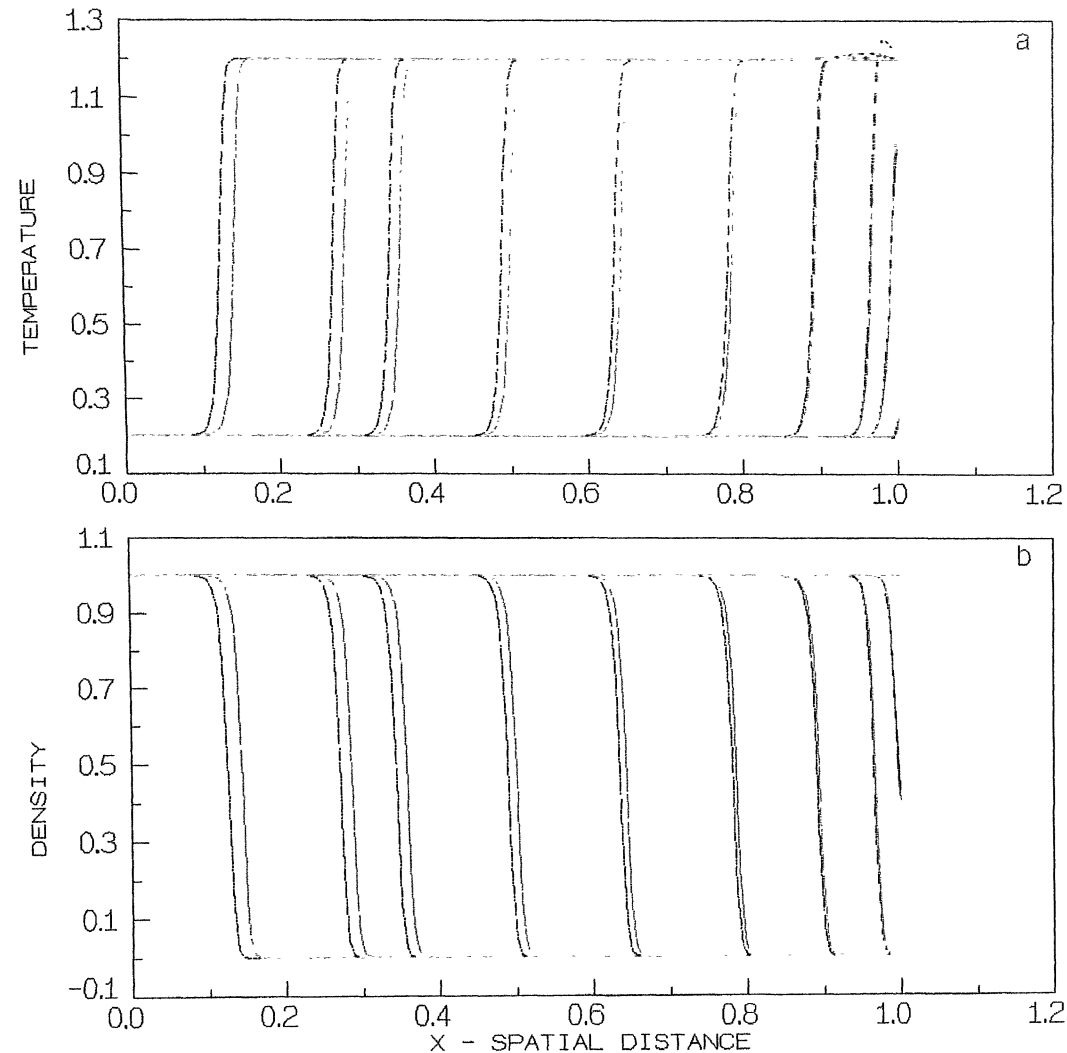
(a). The temperature and the density profiles :

The solution profiles obtained by h version and the uniform mesh $N = 320$, have been superimposed in the Figs. 4.22a-b. Further, a comparison between the solutions obtained by the h and hp version has also been shown in the Figs. 4.23a-b. In both the figures, the solutions were obtained after satisfying the same tolerance limit on the *a posteriori* error. The h and hp computational procedures were started with a coarse mesh of 10 uniform elements. The profiles predicted by the h version and the $N = 320$ meshes are similar. However, there is a small difference in the locations of the flame fronts indicating a slightly inaccurate prediction of flame speed by the uniform mesh. Such a difference is expected since the uniform mesh does

not employ the mesh points optimally and also it does not satisfy the error criterion. The solutions predicted by the h and the h - p strategies are identical at all time instants, thereby verifying the accuracy of each other. The hp method requires less number of elements for the same computable accuracy, but it involves more adaptive cycles. The above comparisons establish the supremacy of the error based adaptive solution strategies over the fixed grid numerical computations. The important point to be noted here is that for the uniform element solution one needs to know the number of elements which will produce solution profiles up to the desired accuracy; for the h and hp methods, the number of elements is automatically decided - but the adequate tolerance level is to be found by trial and error. In the error based methods, however, one has direct control over the solution error and the grid distribution is also more rational.

(b). The behaviour of a posteriori error :

In Fig. 4.24, the variations of a posteriori error and the L_2 norm for the computed gradient h and hp grid refinement strategies have been plotted. The initial number of elements have been taken as 10 in both cases. Because of the coarse initial mesh, the error shoots up to a very high value in the first time step for both methods; but due to the error based adaption strategy, it is brought down rapidly as well, within a single time step. As ignition proceeds, the a posteriori error slightly increases and reaches a peak value with the occurrence of the hump in the temperature profile. As explained in the case of uniform meshes, this peak in the error is linked to the highest rate of reaction



| Mesh | Line_type |
|-----------|-----------|
| h version | ----- |
| N = 320 | ----- |

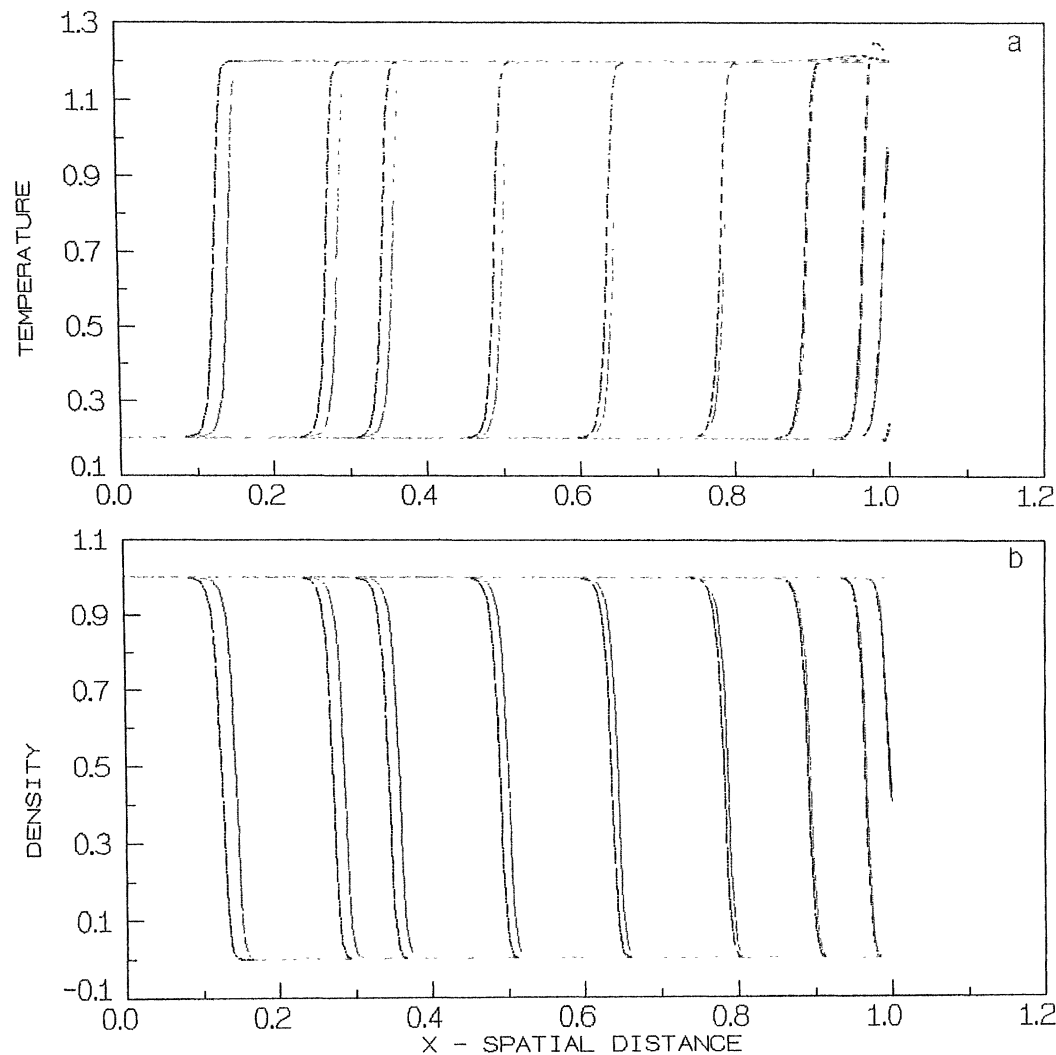
h version mesh

| Adp | Time | Eles | Nods |
|-----|-----------|------|------|
| 1 | 0 .000010 | 12 | 13 |
| 2 | 0 .000150 | 141 | 142 |
| 3 | 0 .000300 | 352 | 353 |
| 4 | 0 .000800 | 349 | 350 |
| 5 | 0 .001550 | 361 | 362 |
| 6 | 0 .002550 | 369 | 370 |
| 7 | 0 .003550 | 329 | 330 |
| 8 | 0 .004550 | 399 | 400 |
| 9 | 0 .005050 | 370 | 371 |
| 10 | 0 .006050 | 355 | 356 |

Uniform Mesh

| Adp | Time | Eles | Nods |
|-----|-----------|------|------|
| 1 | 0 .000010 | 320 | 321 |
| 2 | 0 .000150 | 320 | 321 |
| 3 | 0 .000300 | 320 | 321 |
| 4 | 0 .000800 | 320 | 321 |
| 5 | 0 .001550 | 320 | 321 |
| 6 | 0 .002550 | 320 | 321 |
| 7 | 0 .003550 | 320 | 321 |
| 8 | 0 .004550 | 320 | 321 |
| 9 | 0 .005050 | 320 | 321 |
| 10 | 0 .006050 | 320 | 321 |

Fig. 4.22 FLAME PROBLEM (h & Uniform Mesh) : SOLUTION PROFILE



| Mesh | Line_type |
|------------|-------------|
| h version | ----- |
| hp version | - . - . - . |

h version mesh

| Adp | Time | Eles | Nods |
|-----|-----------|------|------|
| 1 | 0 .000010 | 12 | 13 |
| 2 | 0 .000150 | 141 | 142 |
| 3 | 0 .000300 | 352 | 353 |
| 4 | 0 .000800 | 349 | 350 |
| 5 | 0 .001550 | 361 | 362 |
| 6 | 0 .002550 | 369 | 370 |
| 7 | 0 .003550 | 329 | 330 |
| 8 | 0 .004550 | 399 | 400 |
| 9 | 0 .005050 | 370 | 371 |
| 10 | 0 .006050 | 355 | 356 |

Uniform Mesh

| Adp | Time | Eles | Nods |
|-----|-----------|------|------|
| 1 | 0 .000010 | 320 | 321 |
| 2 | 0 .000150 | 320 | 321 |
| 3 | 0 .000300 | 320 | 321 |
| 4 | 0 .000800 | 320 | 321 |
| 5 | 0 .001550 | 320 | 321 |
| 6 | 0 .002550 | 320 | 321 |
| 7 | 0 .003550 | 320 | 321 |
| 8 | 0 .004550 | 320 | 321 |
| 9 | 0 .005050 | 320 | 321 |
| 10 | 0 .006050 | 320 | 321 |

Fig. 4.23 FLAME PROBLEM (h & Uniform Mesh) : SOLUTION PROFILE

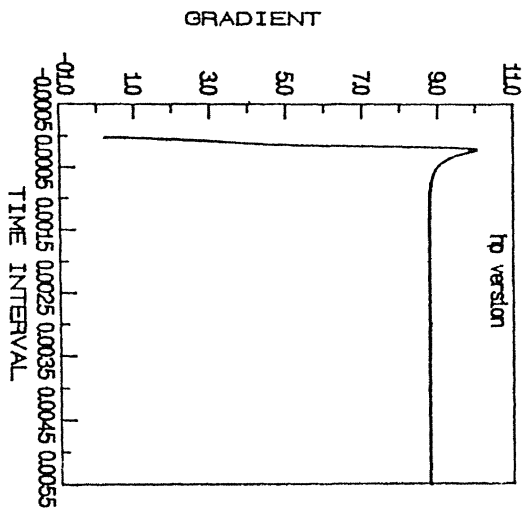
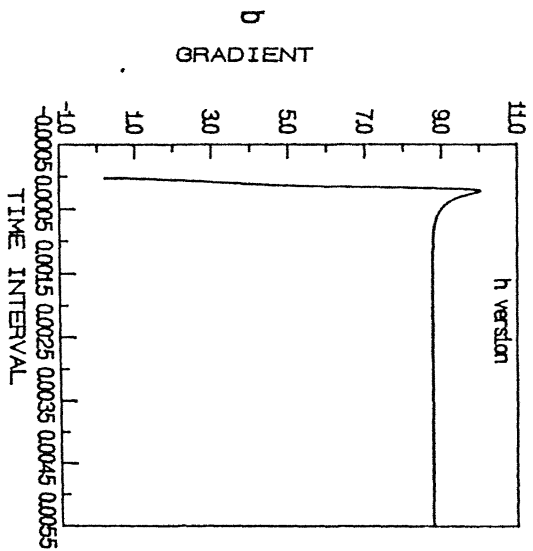
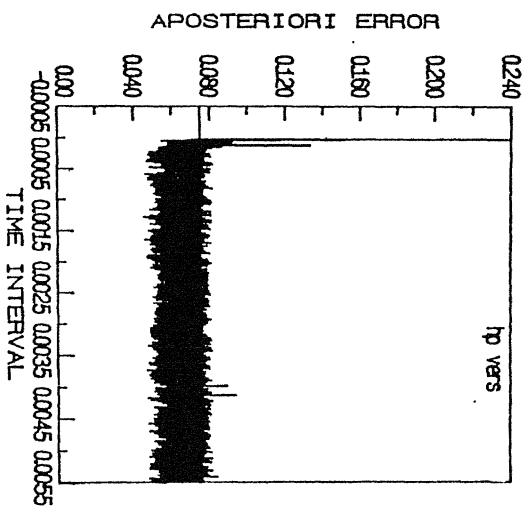
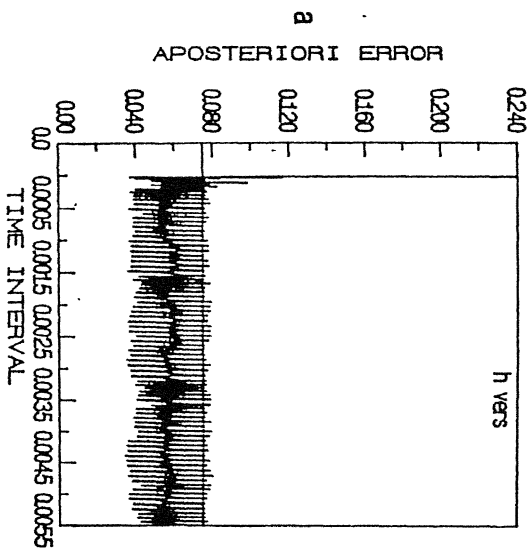


Fig. 4.24. FLAME PROBLEM : TIME EVOLUTION OF ERRORS (h and hp version)
(APOSTERIORI ERROR AND GRADIENT)



prevailing at this juncture. As soon as the flame attains the equilibrium phase of uniform propagation, the *a posteriori* error fluctuates, with a steady mean value. The decrease in the *a posteriori* error seen in the figure after each adaption at any time is due to the effect of the statistical procedure which reduces the local residue in the greater solution activity region. Also, the L_2 norm of the computed gradient is constant in the wave front region except at initial stages for both the methods.

Even though the necessary mathematical theory has not been developed for the implementation of the hp method for the flame propagation problem, the computational experience shows that the present *a posteriori* error estimate for the linear problems works well. The number of adaptations required for the hp strategy is larger due to slower sensitivity of error to grid adaption, as compared to the h version procedure. But with such minor differences excluded, both the methods perform very well in controlling the solution error within the specified tolerance. Most of the trends exhibited by the *a posteriori* error during the application of the adaptive h and hp strategies for the flame propagation problem bear similarities with the uniform mesh results of the flame propagation as well as the test problem results with uniform and non-uniform grids. This indicates that a *a posteriori* error which is based on the residue, is a rugged error measure suitable for grid adaption purposes in both linear and non-linear problems involving front propagation. Some of the numerical results obtained by the h and hp refinement methods for various time instants have also been presented in the tables (4.5) and (4.6a-b).

| Time | Elments | | | | | | Apos.err | Com.grd |
|----------|---------|-----|-----|-----|-----|-------|----------|---------|
| | Adp | Rem | Ref | Stf | N | Nodes | | |
| 0.000010 | 1 | 0 | 0 | 0 | 10 | 11 | 24.52254 | 0.2078 |
| | 2 | 1 | 1 | 4 | 12 | 13 | 3.06718 | 0.4067 |
| | 3 | 2 | 1 | 4 | 13 | 14 | 0.40373 | 0.6596 |
| | 4 | 5 | 1 | 4 | 11 | 12 | 0.08006 | 0.7418 |
| | 5 | 3 | 2 | 6 | 12 | 13 | 0.03707 | 0.7495 |
| 0.000020 | 1 | 0 | 0 | 0 | 12 | 13 | 0.11114 | 1.1890 |
| | 2 | 0 | 2 | 6 | 16 | 17 | 0.11693 | 1.1835 |
| | 3 | 0 | 3 | 10 | 23 | 24 | 0.05885 | 1.1824 |
| 0.000060 | 1 | 0 | 0 | 0 | 31 | 32 | 0.07549 | 2.6877 |
| | 2 | 2 | 8 | 20 | 41 | 42 | 0.06521 | 2.6872 |
| 0.000115 | 1 | 0 | 0 | 0 | 59 | 60 | 0.07878 | 4.4325 |
| | 2 | 3 | 15 | 36 | 77 | 78 | 0.05337 | 4.4327 |
| 0.000145 | 1 | 0 | 0 | 0 | 113 | 114 | 0.07601 | 6.4919 |
| | 2 | 1 | 23 | 52 | 141 | 142 | 0.05562 | 6.4918 |
| 0.000160 | 1 | 0 | 0 | 0 | 141 | 142 | 0.08230 | 8.1623 |
| | 2 | 1 | 26 | 58 | 172 | 173 | 0.06618 | 8.1623 |
| 0.000200 | 1 | 0 | 0 | 0 | 232 | 233 | 0.07582 | 10.0157 |
| | 2 | 32 | 72 | 186 | 314 | 315 | 0.03995 | 10.0156 |
| 0.000245 | 1 | 0 | 0 | 0 | 314 | 315 | 0.07713 | 9.8579 |
| | 2 | 72 | 73 | 182 | 351 | 352 | 0.03751 | 9.8578 |
| 0.000360 | 1 | 0 | 0 | 0 | 352 | 353 | 0.07791 | 9.2782 |
| | 2 | 95 | 54 | 144 | 347 | 348 | 0.03911 | 9.2781 |
| 0.000500 | 1 | 0 | 0 | 0 | 346 | 347 | 0.07586 | 8.9779 |
| | 2 | 95 | 77 | 196 | 370 | 371 | 0.03682 | 8.9777 |
| 0.000855 | 1 | 0 | 0 | 0 | 349 | 350 | 0.07583 | 8.7898 |
| | 2 | 93 | 57 | 138 | 337 | 338 | 0.04011 | 8.7896 |
| 0.001165 | 1 | 0 | 0 | 0 | 360 | 361 | 0.07806 | 8.7668 |
| | 2 | 97 | 57 | 150 | 356 | 357 | 0.03791 | 8.7666 |
| 0.001670 | 1 | 0 | 0 | 0 | 351 | 352 | 0.07850 | 8.7621 |
| | 2 | 95 | 71 | 146 | 331 | 332 | 0.04474 | 8.7619 |
| 0.002635 | 1 | 0 | 0 | 0 | 367 | 368 | 0.07552 | 8.7614 |
| | 2 | 99 | 60 | 146 | 354 | 355 | 0.03883 | 8.7612 |
| 0.003355 | 1 | 0 | 0 | 0 | 333 | 334 | 0.07675 | 8.7616 |
| | 2 | 90 | 75 | 154 | 322 | 323 | 0.04212 | 8.7614 |
| 0.004855 | 1 | 0 | 0 | 0 | 380 | 381 | 0.07877 | 8.7623 |
| | 2 | 108 | 54 | 122 | 340 | 341 | 0.04269 | 8.7621 |
| 0.005810 | 1 | 0 | 0 | 0 | 347 | 348 | 0.07504 | 8.7620 |
| | 2 | 93 | 65 | 164 | 353 | 354 | 0.03976 | 8.7618 |
| 0.005960 | 1 | 0 | 0 | 0 | 351 | 352 | 0.07687 | 8.7621 |
| | 2 | 97 | 62 | 156 | 348 | 349 | 0.03796 | 8.7619 |
| 0.006035 | 1 | 0 | 0 | 0 | 348 | 349 | 0.07728 | 8.7625 |
| | 2 | 96 | 65 | 168 | 355 | 356 | 0.03686 | 8.7623 |

Table. 4.5. h version results for flame propagation problem

| Time | Adp | Deg | Rem | N | Dof | Apos.err | Com.grd |
|----------|-----|-----|-----|-----|-----|----------|---------|
| 0.000010 | 1 | 1 | 0 | 10 | 11 | 24.52254 | 0.2078 |
| | 2 | 1 | 1 | 9 | 10 | 24.52254 | 0.2078 |
| | 3 | 2 | 0 | 9 | 11 | 8.17348 | 0.2881 |
| | 4 | 3 | 0 | 9 | 12 | 3.95437 | 0.3722 |
| | 5 | 4 | 0 | 9 | 13 | 2.23357 | 0.4538 |
| | 6 | 5 | 0 | 9 | 14 | 1.34661 | 0.5292 |
| | 7 | 1 | 0 | 12 | 13 | 0.48328 | 0.6393 |
| | 8 | 1 | 3 | 9 | 10 | 0.48328 | 0.6393 |
| | 9 | 1 | 2 | 7 | 8 | 0.48328 | 0.6393 |
| | 10 | 2 | 0 | 7 | 9 | 0.15986 | 0.7236 |
| | 11 | 3 | 0 | 7 | 10 | 0.09315 | 0.7465 |
| | 12 | 4 | 0 | 7 | 11 | 0.08333 | 0.7491 |
| | 13 | 5 | 0 | 7 | 12 | 0.07093 | 0.7493 |
| 0.000040 | 1 | 1 | 0 | 25 | 26 | 0.08108 | 1.9831 |
| | 2 | 1 | 2 | 23 | 24 | 0.08205 | 1.9832 |
| | 3 | 2 | 0 | 23 | 30 | 0.05484 | 1.9834 |
| 0.000110 | 1 | 1 | 0 | 43 | 44 | 0.08978 | 4.2384 |
| | 2 | 2 | 0 | 43 | 56 | 0.13341 | 4.2382 |
| | 3 | 3 | 0 | 43 | 57 | 0.12053 | 4.2382 |
| | 4 | 4 | 0 | 43 | 58 | 0.11326 | 4.2382 |
| | 5 | 5 | 0 | 43 | 59 | 0.10735 | 4.2382 |
| | 6 | 1 | 0 | 47 | 48 | 0.08258 | 4.2383 |
| | 7 | 1 | 1 | 46 | 47 | 0.08259 | 4.2383 |
| | 8 | 2 | 0 | 46 | 60 | 0.06729 | 4.2382 |
| 0.000255 | 1 | 1 | 0 | 236 | 237 | 0.07992 | 9.7930 |
| | 2 | 1 | 10 | 226 | 227 | 0.08011 | 9.7930 |
| | 3 | 1 | 1 | 225 | 226 | 0.08013 | 9.7930 |
| | 4 | 2 | 0 | 225 | 278 | 0.05139 | 9.7930 |
| 0.000790 | 1 | 1 | 0 | 211 | 212 | 0.07918 | 8.8032 |
| | 2 | 1 | 10 | 201 | 202 | 0.07941 | 8.8032 |
| | 3 | 1 | 1 | 200 | 201 | 0.07945 | 8.8032 |
| | 4 | 2 | 0 | 200 | 246 | 0.05102 | 8.8032 |
| 0.001270 | 1 | 1 | 0 | 221 | 222 | 0.07489 | 8.7648 |
| | 2 | 1 | 14 | 207 | 208 | 0.07528 | 8.7647 |
| | 3 | 1 | 1 | 206 | 207 | 0.07531 | 8.7647 |
| | 4 | 2 | 0 | 206 | 268 | 0.04535 | 8.7648 |
| 0.001500 | 1 | 1 | 0 | 220 | 221 | 0.07531 | 8.7622 |
| | 2 | 1 | 7 | 213 | 214 | 0.07545 | 8.7622 |
| | 3 | 1 | 1 | 212 | 213 | 0.07547 | 8.7622 |
| | 4 | 2 | 0 | 212 | 263 | 0.04794 | 8.7623 |
| 0.001720 | 1 | 1 | 0 | 217 | 218 | 0.07503 | 8.7612 |
| | 2 | 1 | 12 | 205 | 206 | 0.07529 | 8.7612 |
| | 3 | 2 | 0 | 205 | 256 | 0.04864 | 8.7612 |
| 0.002020 | 1 | 1 | 0 | 209 | 210 | 0.08069 | 8.7614 |
| | 2 | 1 | 11 | 198 | 199 | 0.08095 | 8.7614 |
| | 3 | 2 | 0 | 198 | 249 | 0.05474 | 8.7614 |

Table. 4.6a. h-p version results for flame propagation problem

(c). The effect on Maximum and Mean of local error indicators :

The plots of maximum and mean local error variations with respect to time are presented in Figs. 4.25 a-b, for the h and hp methods. The difference between the maximum and the mean errors quantifies the extent of adaption required at the location of the maximum error. At initial times (during the ignition phase), the difference between the maximum and the mean errors is high, indicating the great need for grid adaption. Indeed, at this time, the mesh structure changes from coarse to fine by rapid stuffing and removal of elements. After the completion of the ignition phase (and the hump region), the difference between the maximum and the mean errors reduces considerably, indicating a well graded mesh. Occasionally, due to slight mismatch between the existing mesh and the front location, the maximum error exhibits high peaks. The mean local error, on the other hand, has a steady value during equilibrium propagation phase. The peaks in the maximum error are followed by sharp falls due to the grid adaption process which sets in at these peak points. In fact, after each adaption the maximum error is brought close to the mean value by equidistributing the error, which indicates the efficient application of the statistical procedure. Due to the high sensitivity of the h version, the difference between the maximum and the mean becomes very small after only one grid adaption. For the hp method, in fact, the statistical procedure is not as effective as in the h method. The number of adaptations required in hp version required is also much more as discussed earlier. Due to this large number of grid adaptations and also the conversions

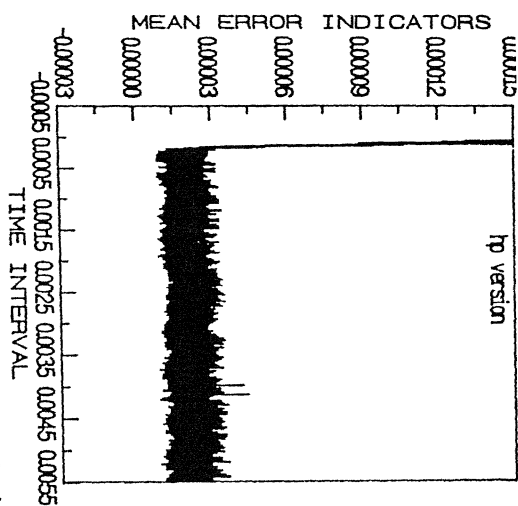
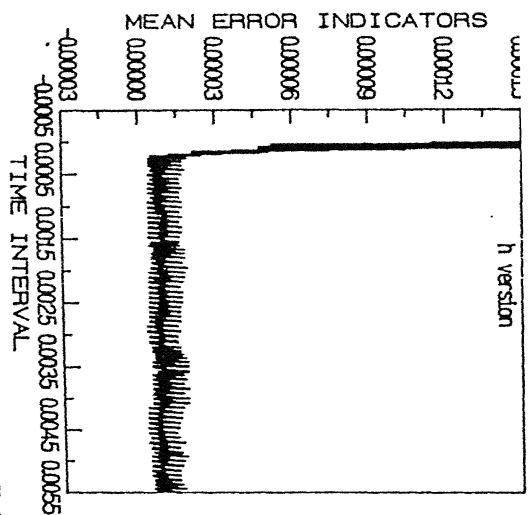
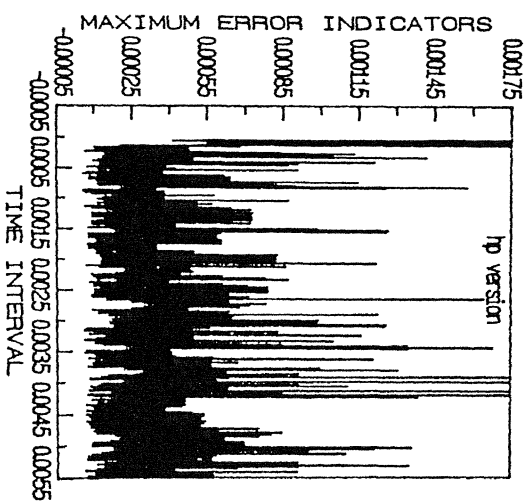
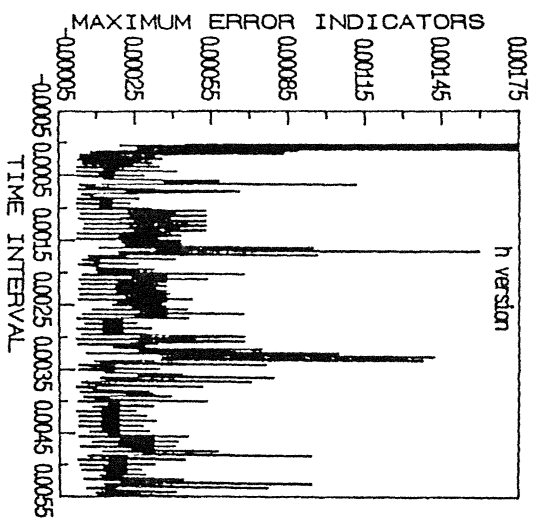


Fig. 4.25. FLAME PROBLEM : TIME EVOLUTION OF ERRORS (h and hp version)
(MAXIMUM AND MEAN ERROR INDICATORS)



from p to h-version meshes, the mean for the hp error forms a thick band with respect to time. In general, both the mean and the maximum of local errors for the h method are below those of the hp method. Most of these features of local errors for the flame problem are similar to the trends observed for the test problem when the non-uniform meshes are used.

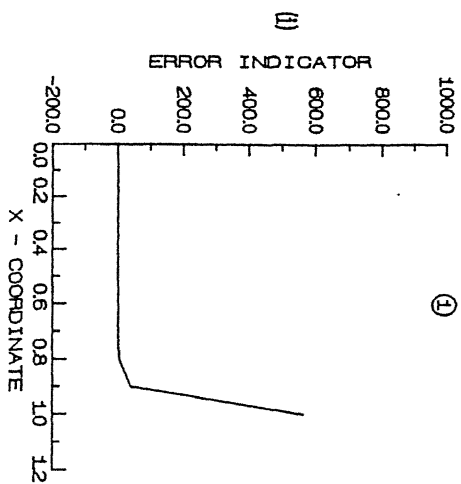
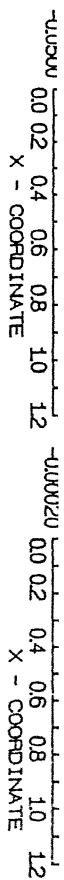
(d). Behaviour of local error indicators and distribution :

The distributions of the square of local error indicators at various time instants and adaption levels have been plotted in Figs. 4.26a-f for the h refinement method. The time levels have been chosen such that the ignition, flame propagation and extinction phases of the flame are all covered. The computations have not been performed till extinction is complete. With a starting mesh of 10 uniform elements, the error indicator is very large in magnitude adjacent to the boundary where the flame ignites (see Fig. 4.26a). As the mesh is refined according to the statistical procedure in the large error zone, the error indicator decreases rapidly. Also, in the hump region the of solution (i.e. near $x = 1.0$ boundary at $t = 0.0003$), the behaviour of the local error indicators exhibit fluctuations as seen from Fig. 4.26b. By examining the ratios of the maximum error and the corresponding step length the decrease in error after grid adaption is observed to be proportional to h. It is to be noted, however, that the precise dependence of the error indicators on the local step length is very difficult to ascertain due to the fact that the predominant contribution to the local residue arises from the non-linear source term. This aspect is

addressed in greater detail in Chapter 5. In Fig. 4.26a, it is seen that the change in the number of elements after each adaption is very small, although the local error has decreased by orders of magnitude. The reason for this is the simultaneous removal of nodes from unwanted zones, along with grid refinement.

At different time instants, the local error distribution exhibits similar features. The maximum value of the local error occurs at the location of maximum reaction rate (flame front region). Occasionally, due to excessive coarseness of the mesh, the local error increases for some adaptations in the non-reaction zones also. However, this increase is confined to one or two elements only and gets immediately rectified by proper grid-spacing adjustment in the subsequent adaptations (see, Fig. 4.26b-c). Thus, with regard to the satisfaction of the global error criterion, these large elements do not have significant influence. In general, the magnitude of the maximum local error is brought down considerably (by almost an order of magnitude) by each h version grid-adaption.

The width of the flame zone can be correlated with the region where close peaks in the local error are observed. It is clear from the Figs. (4.26 a-f) that during the ignition phase the flame region is very thin but its width grows with time and reaches a constant value during the equilibrium propagation phase. Due to this increase in width, more number of elements contribute to global error after the ignition phase. Therefore, if the same global error value is maintained for all times, the satisfaction of error tolerance criterion leads to very small values of the local error for later time instants, than in the beginning. This



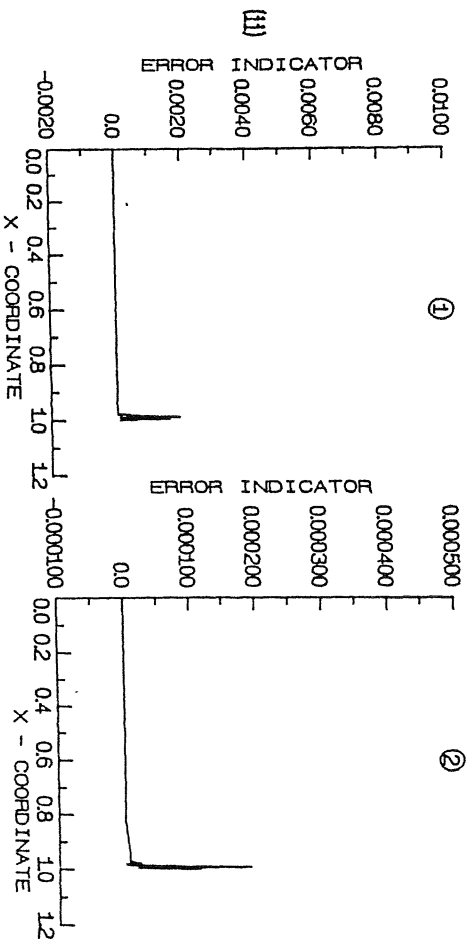
Graph (1)

| Rdp | Nem | Nod |
|-----|-----|-----|
| 1 | 10 | 11 |

Graph (2)

| Rdp | Nem | Nod |
|-----|-----|-----|
| 2 | 12 | 13 |

Fig. 4.26a. FLAME PROBLEM : BEHAVIOUR OF ERRORS (h VERSION)
(I), $t = 0.00001$, (II), $t = 0.00003$

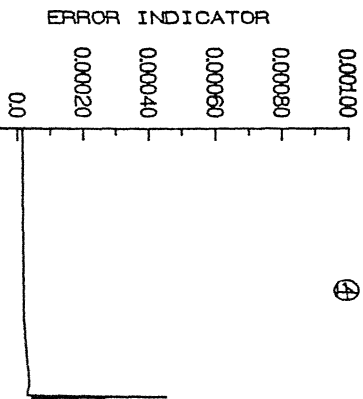
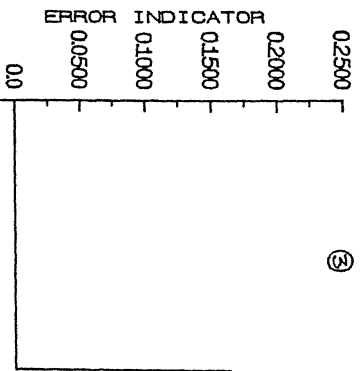


Graph (1)

| Rdip | Nem | Nod |
|------|-----|-----|
| 1 | 23 | 24 |

Graph (2)

| Rdip | Nem | Nod |
|------|-----|-----|
| 2 | 31 | 32 |



Graph (3)

| Rdip | Nem | Nod |
|------|-----|-----|
| 3 | 13 | 14 |

Graph (4)

| Rdip | Nem | Nod |
|------|-----|-----|
| 4 | 12 | 13 |

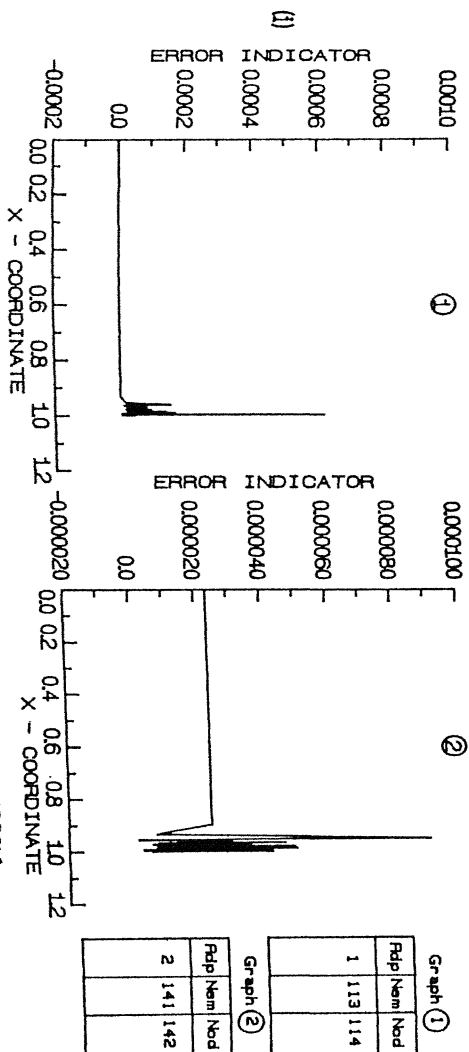
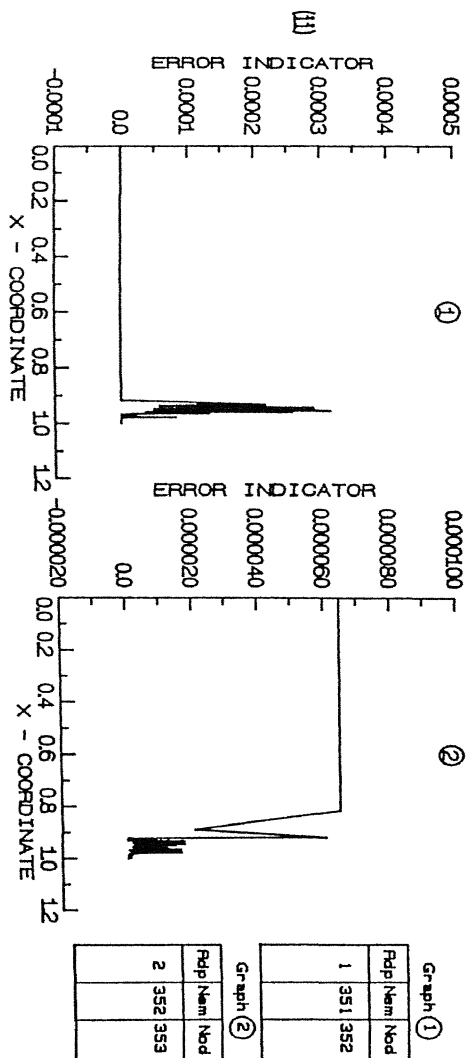
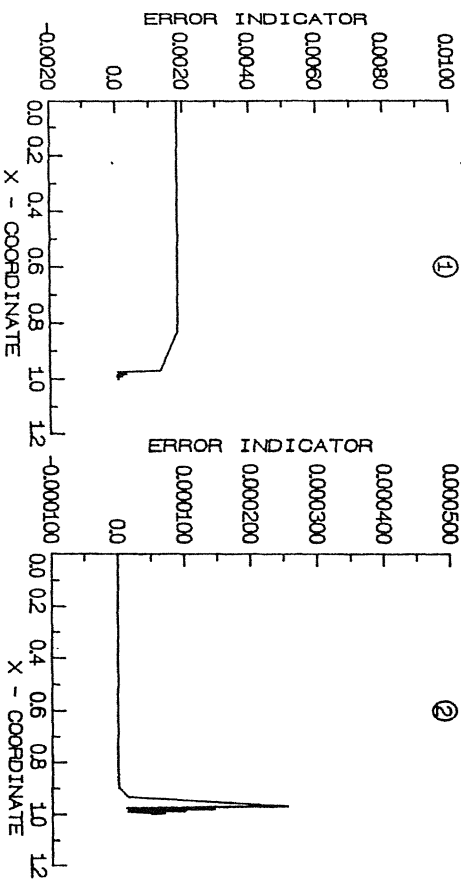


Fig. 4.26b. FLAME PROBLEM : BEHAVIOUR OF ERRORS (h VERSION)
 (i). $t = 0.000145$, (ii). $t = 0.00030$, (iii). $t = 0.00070$

(III)



Graph ①

| Pdip | Nem | Nod |
|------|-----|-----|
| 1 | 41 | 42 |

Graph ②

| Pdip | Nem | Nod |
|------|-----|-----|
| 2 | 50 | 51 |

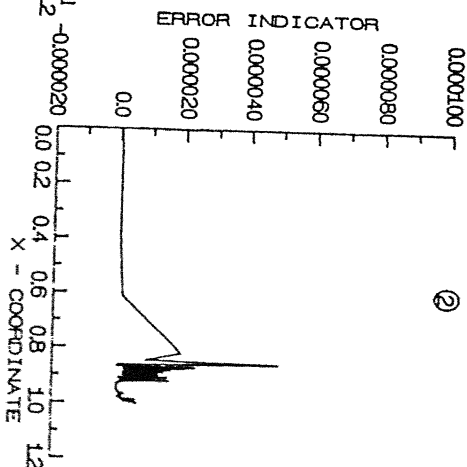
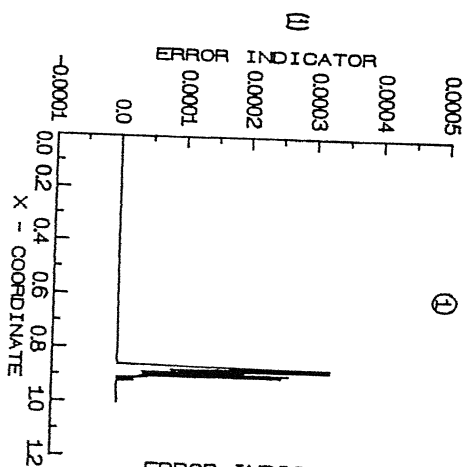


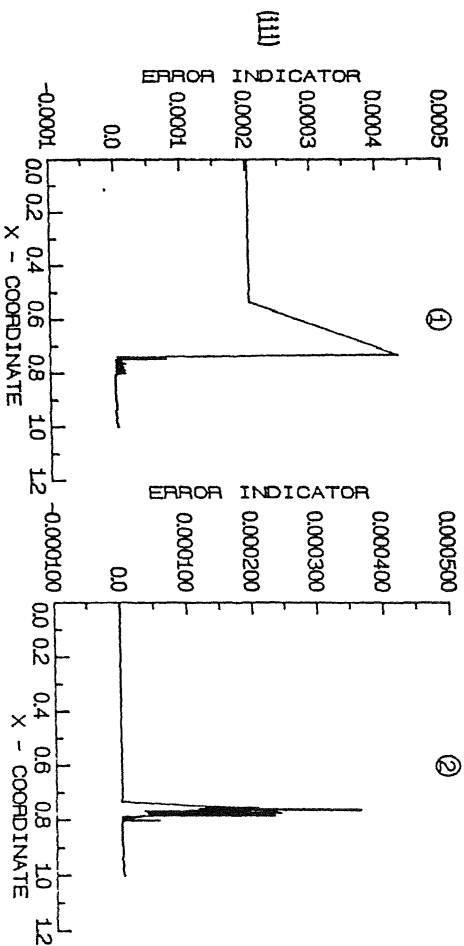
Fig. 4.26c. FLAME PROBLEM : BEHAVIOUR OF ERRORS (h VERSION)
 (1). $t = 0.00071$, (2). $t = 0.00105$, (3). $t = 0.00155$

Graph (1)

| RdP | Nem | Nod |
|-----|-----|-----|
| 1 | 352 | 353 |

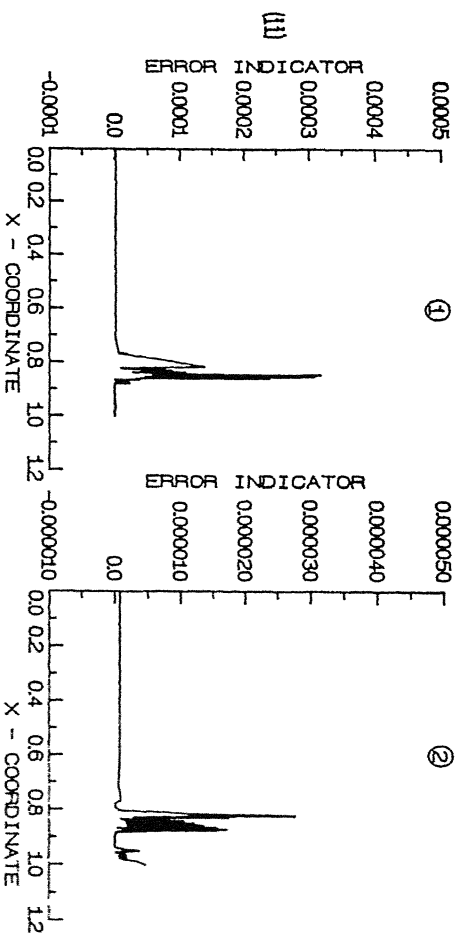
Graph (2)

| RdP | Nem | Nod |
|-----|-----|-----|
| 2 | 347 | 348 |



| Graph ① | |
|---------|---------|
| Rdp | Nom Nod |
| 1 | 362 363 |
| | |
| | |

| Graph ② | |
|---------|---------|
| Rdp | Nom Nod |
| 2 | 361 362 |
| | |
| | |



| Graph ① | |
|---------|---------|
| Rdp | Nom Nod |
| 1 | 349 350 |
| | |
| | |

| Graph ② | |
|---------|---------|
| Rdp | Nom Nod |
| 2 | 368 369 |
| | |
| | |

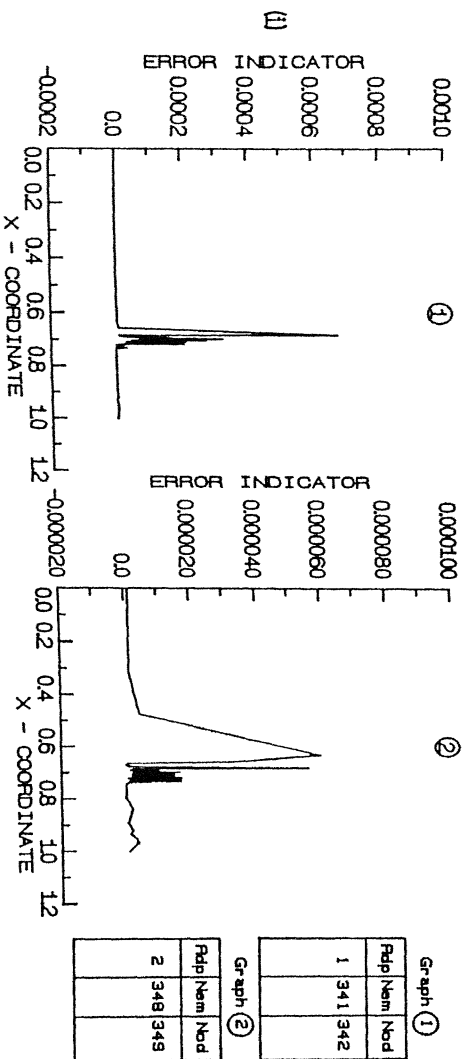
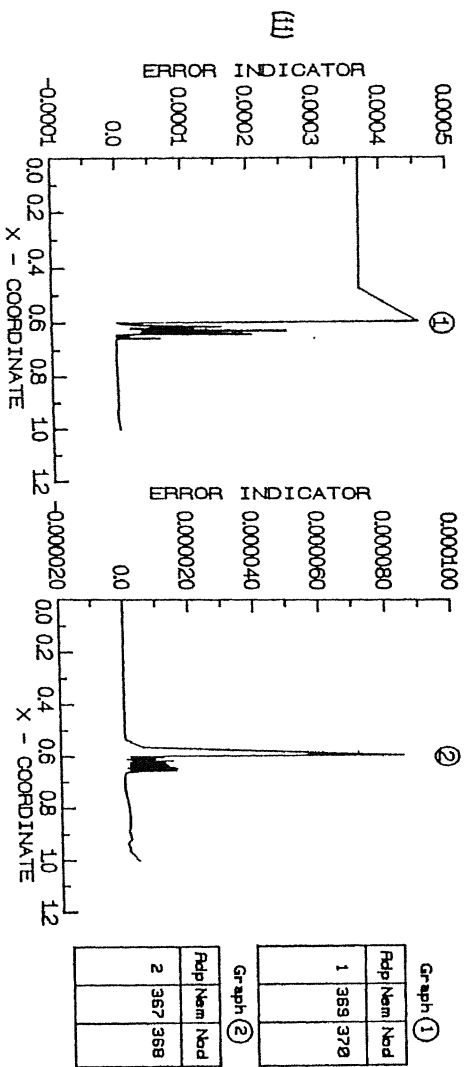
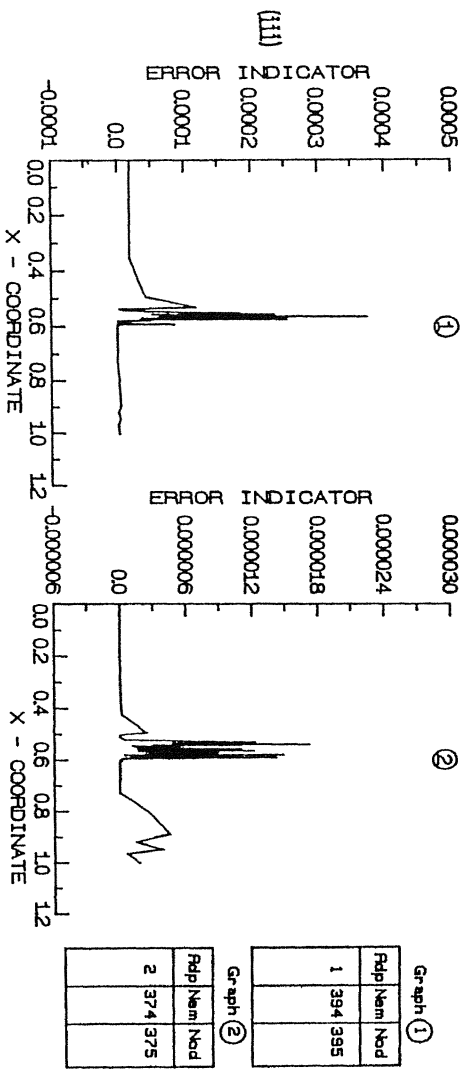
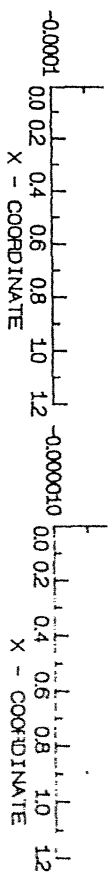


Fig. 4.26d. FLAME PROBLEM : BEHAVIOR OF ERRORS IN VERSION
 (i). $t = 0.002025$, (ii). $t = 0.002565$, (iii). $t = 0.003025$



Graph ①

| Pdp | Nem | Nod |
|-----|-----|-----|
| 1 | 329 | 338 |

Graph ②

| Pdp | Nem | Nod |
|-----|-----|-----|
| 2 | 330 | 331 |

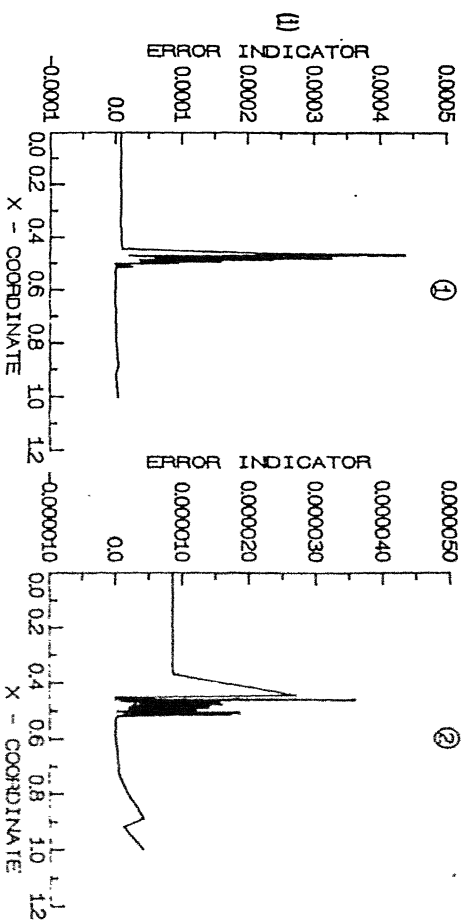
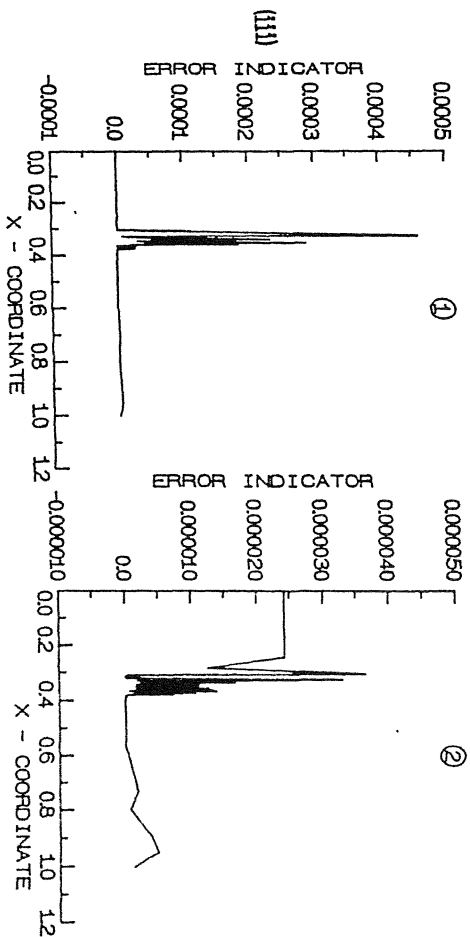


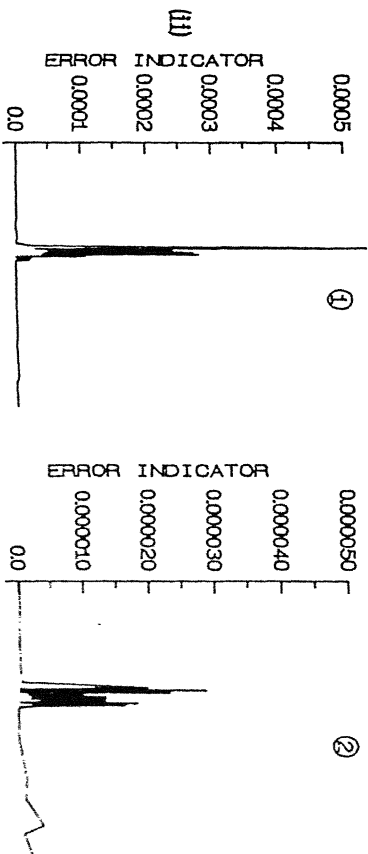
Fig. 4.26e. FLAME PROBLEM : BEHAVIOUR OF ERRORS In VERSION
((I), $t = 0.003570$, (III), $t = 0.003995$, (III), $t = 0.004545$)



| Rdp | Nem | Nod |
|-----|-----|-----|
| 1 | 410 | 411 |
| | | |

Graph ②

| Rdp | Nem | Nod |
|-----|-----|-----|
| 2 | 399 | 400 |
| | | |



| Rdp | Nem | Nod |
|-----|-----|-----|
| 1 | 337 | 338 |
| | | |

Graph ②

| Rdp | Nem | Nod |
|-----|-----|-----|
| 2 | 325 | 326 |
| | | |

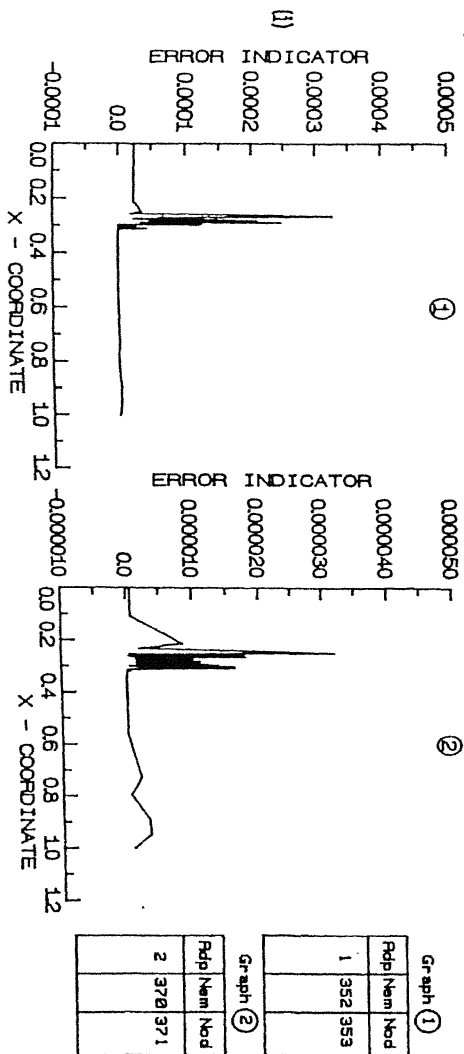
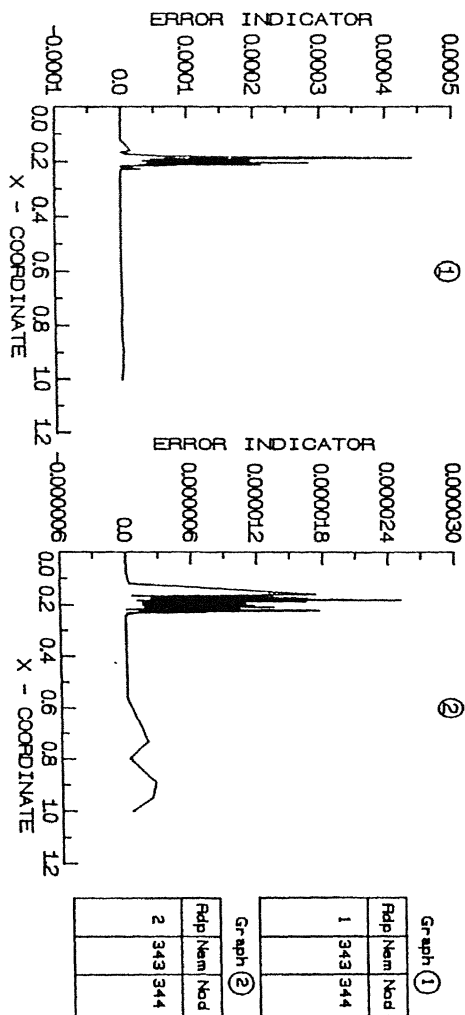
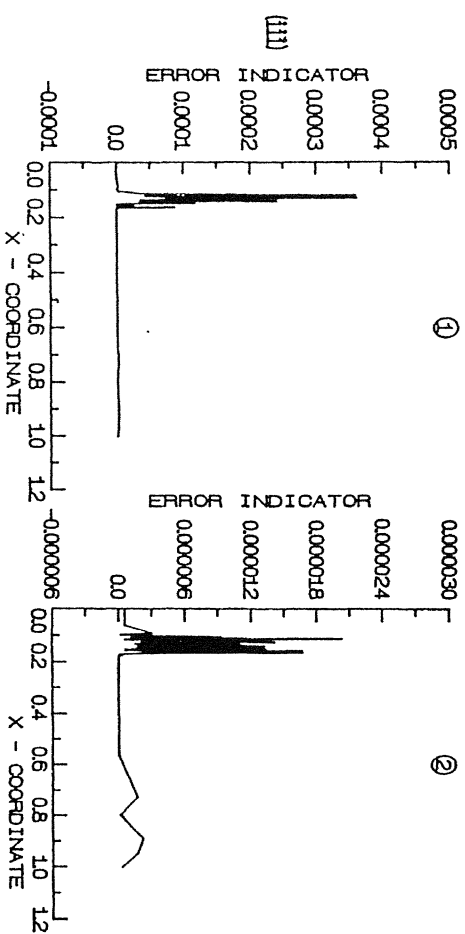


Fig. 4.26f. FLAME PROBLEM : BEHAVIOUR OF ERRORS (n VERSION)
 (i). $t = 0.004995$, (ii). $t = 0.005580$, (iii). $t = 0.006035$



Graph ①

| PdP | Nem | Nod |
|-----|-----|-----|
| 1 | 348 | 349 |
| | | |

Graph ②

| PdP | Nem | Nod |
|-----|-----|-----|
| 2 | 355 | 356 |
| | | |

effect is clearly observed in the figures. For time values greater than 0.0003, the fully developed equilibrium flame front is established and the number of elements does not vary much during this period with respect to time as well as adaptations within a time step. The main change that occurs is the redistribution of dense grid point zone so as to match the shifting location of the flame front region. Because of this, the band of local error peaks observed in the Figs. 4.26a-f, shifts its location with respect to time level or adaption within a time step. As regards the number of adaptations required for satisfying the error criterion, it is high only for the first few time steps and for later times, the error tolerance is met within 2 adaptations, since the grid distribution becomes almost optimal at every stage and deviations from optimality are minimal.

The mesh distribution at different time instants and adaption steps, are shown in Figs. 4.27a-b for the h version method. The solution procedure was started with an initial mesh of 10 uniform elements. In the first time step, simultaneous refinement and element removal (derefinement) are seen during each adaption. It is observed that the refinement procedure is somewhat slower than the derefinement. There are two reasons for this : (1) In the present algorithm when two neighbouring elements have error below a prescribed minimum tolerance level, they are collapsed into a single element. In a given adaption step, this procedure is carried out on every pair of existing elements satisfying the above criterion. However, for the new elements created by the merger of two elements in the existing mesh, the check for element collapse will be performed only in the next adaption. (2) The

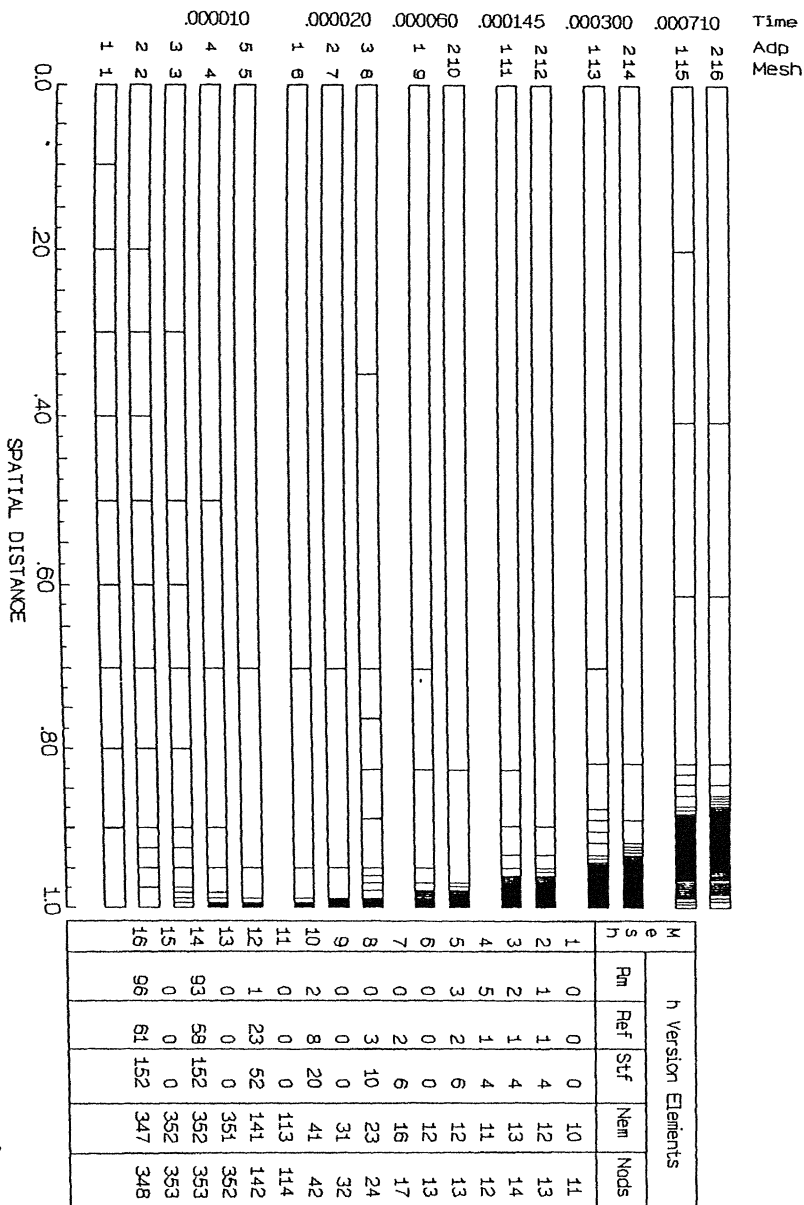


FIG.4.27(d) h - ADAPTED MESH AT VARIOUS TIME INSTANTS (FLAME PROBLEM)

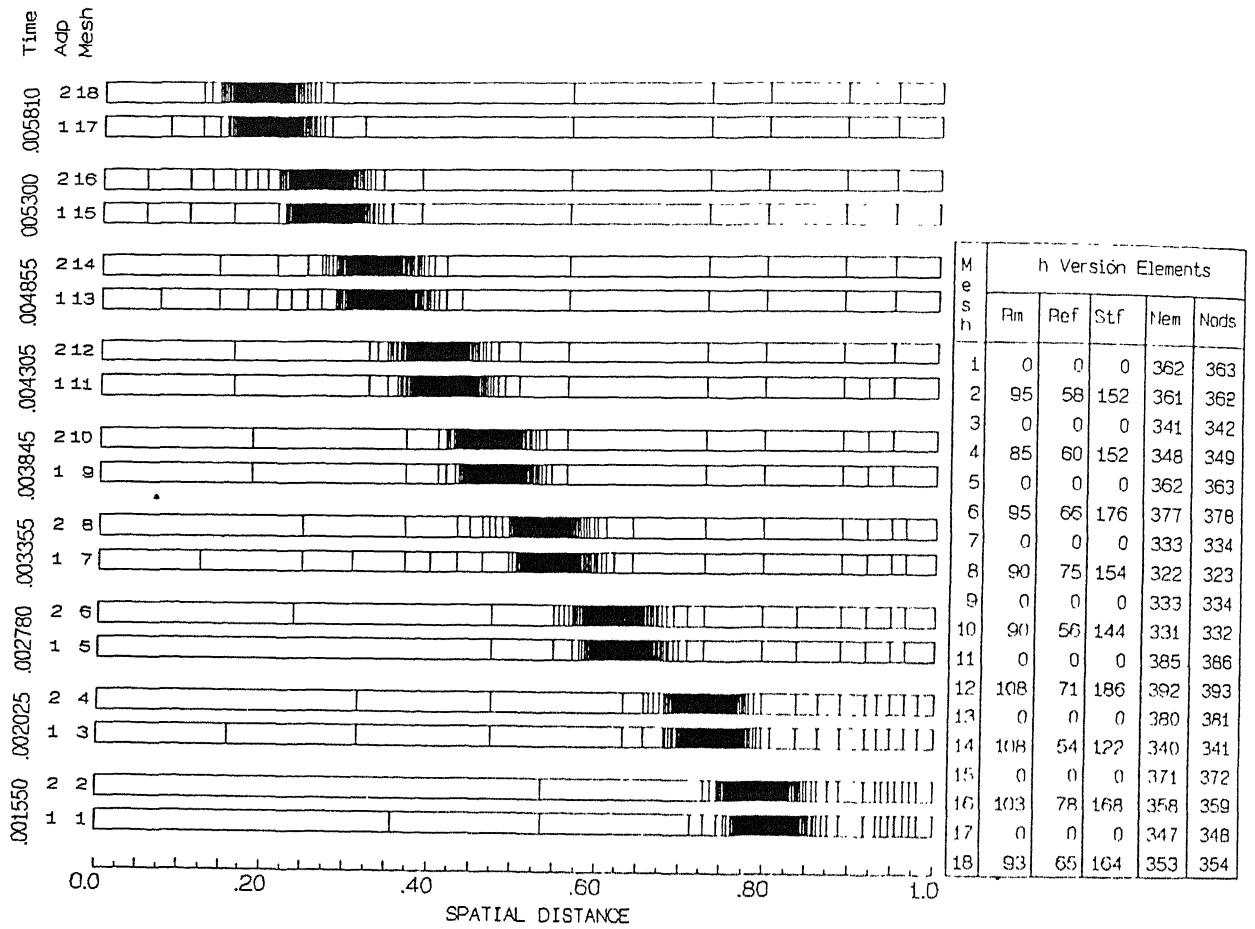
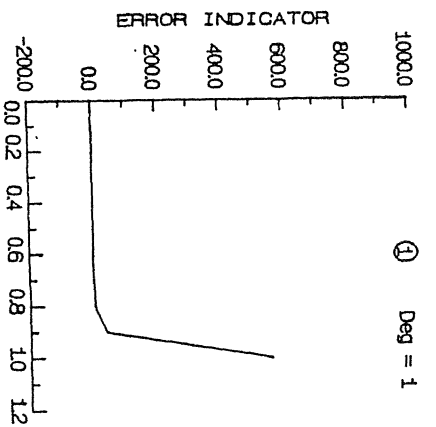


FIG.4.27(b) h - ADAPTED MESH AT VARIOUS TIME INSTANTS (FLAME PROBLEM)

magnitudes of the error indicators and their statistical measures (mean, standard deviation, etc.,) decrease with grid adaption. Thus, elements which would not have qualified for collapse become so in the subsequent adaption steps.

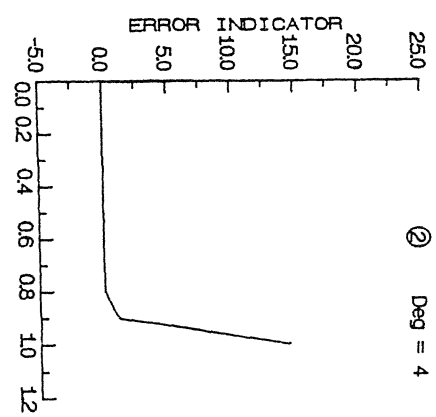
It should be mentioned in passing, however, that the adaption process on the coarse mesh is influenced by refinement or derefinement occurring anywhere in the domain; therefore, execution of removal mechanism for elements in the non reactive region depends on the grid adaption occurring in the other places. These features are clearly seen in the evolution of the adaptive mesh at $t = 0.00002$. With respect to time, it is observed that the numerical grid faithfully follows the front movement. In fact, if the element distribution is not optimal, then within the same time step the mesh is adjusted appropriately in subsequent iterations. It is observed in the figures that during this iteration process, the dense mesh region shifts to the exact location of the flame front. The width of the dense mesh region which is a measure of the thickness of the flame front, remains more or less constant during the equilibrium flame propagation-phase. It is observed that the total number of elements in the solution domain fluctuates with time, even in the equilibrium phase. The reason for this is that the tolerance criterion for error is an inequality constraint only and therefore, any number of elements above a certain minimum value will satisfy the error tolerance.

In Figs. 4.28a-f, the distribution of the squares of the local error indicators have been shown for the hp refinement procedure. The corresponding adaptive meshes and the degree of interpolation used within the elements are depicted in



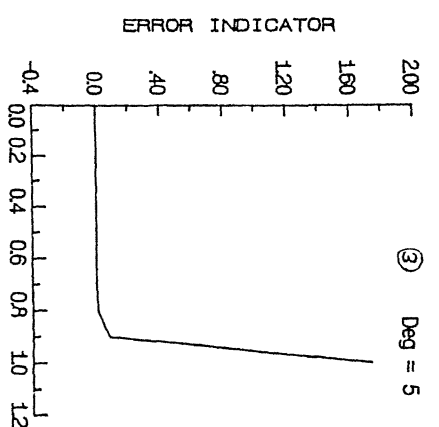
Graph ①

| Adp | Nem | DoF |
|-----|-----|-----|
| 1 | 10 | 11 |
| 2 | 9 | 12 |



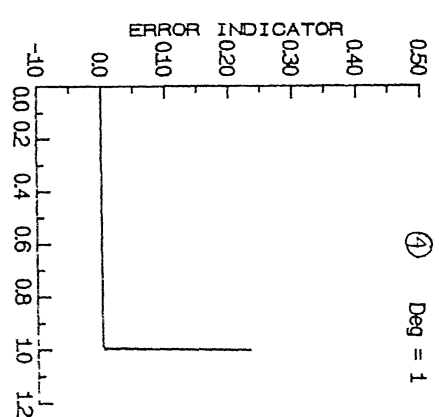
Graph ②

| Adp | Nem | DoF |
|-----|-----|-----|
| 1 | 10 | 11 |
| 2 | 9 | 12 |



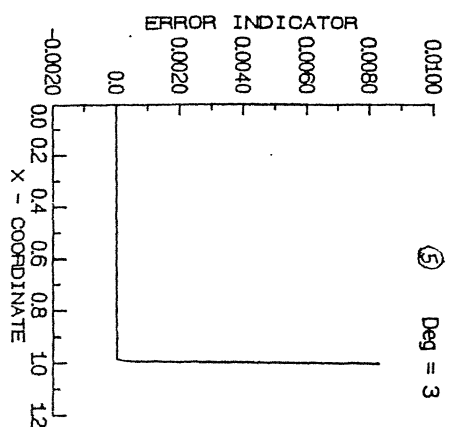
Graph ③

| Adp | Nem | DoF |
|-----|-----|-----|
| 1 | 9 | 14 |
| 2 | 12 | 13 |



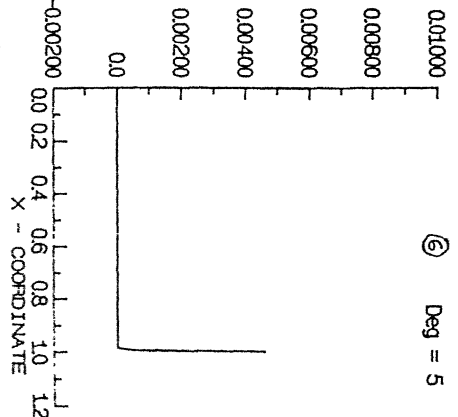
Graph ④

| Adp | Nem | DoF |
|-----|-----|-----|
| 1 | 9 | 14 |
| 2 | 12 | 13 |



Graph ⑤

| Adp | Nem | DoF |
|-----|-----|-----|
| 1 | 7 | 10 |
| 2 | 7 | 12 |



Graph ⑥

| Adp | Nem | DoF |
|-----|-----|-----|
| 1 | 7 | 10 |
| 2 | 7 | 12 |

Fig. 4.28a. FLAME PROBLEM : BEHAVIOUR OF ERRORS (hp VERSION)
($t = 0.00001$)

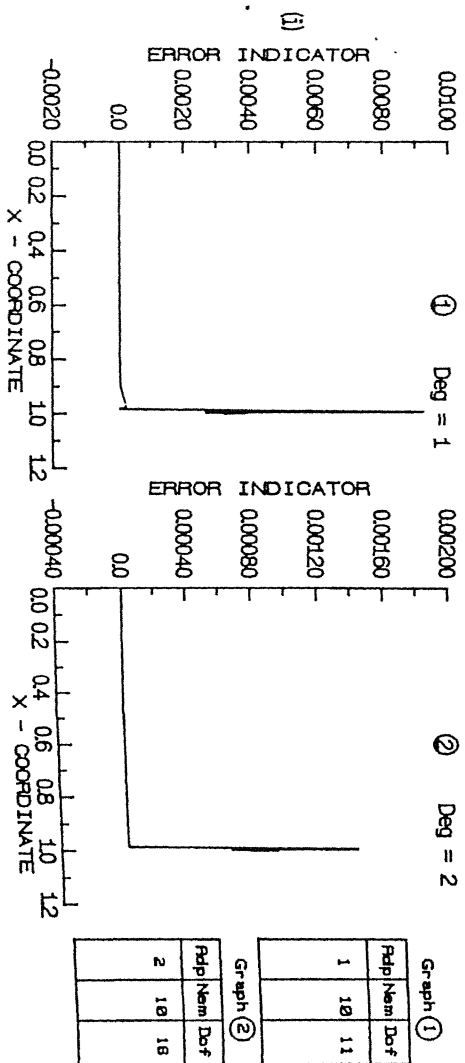
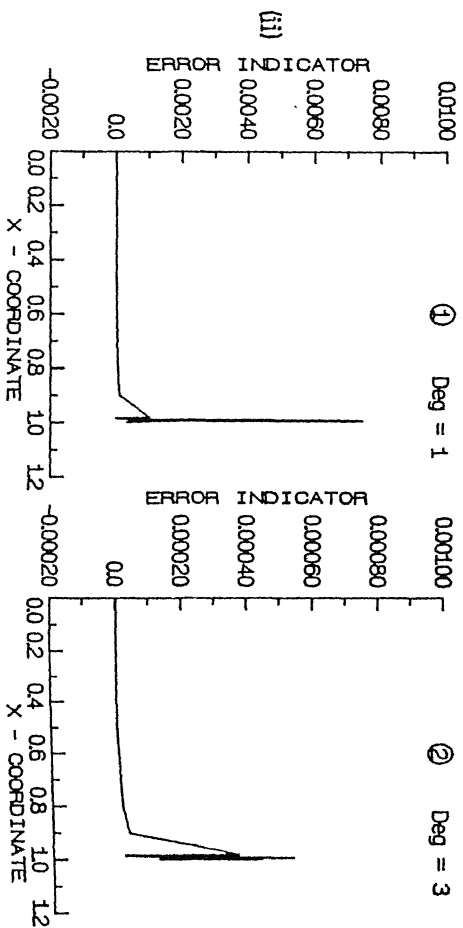


Fig. 4.28b. FLAME PROBLEM : BEHAVIOUR OF ERRORS (hp VERSION)
 (i), $t = 0.000002$, (ii), $t = 0.000003$

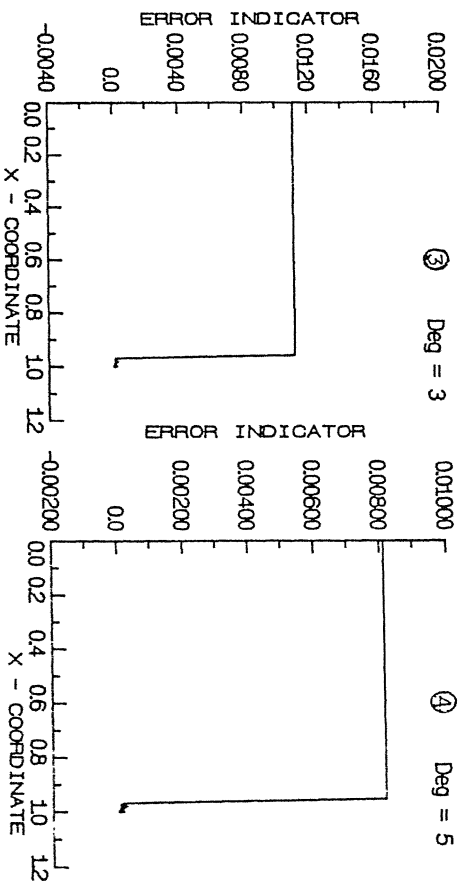


Graph ①

| Pdip Nam | Dof |
|----------|-------|
| 1 | 15 16 |

Graph ②

| Pdip Nam | Dof |
|----------|-------|
| 2 | 15 26 |

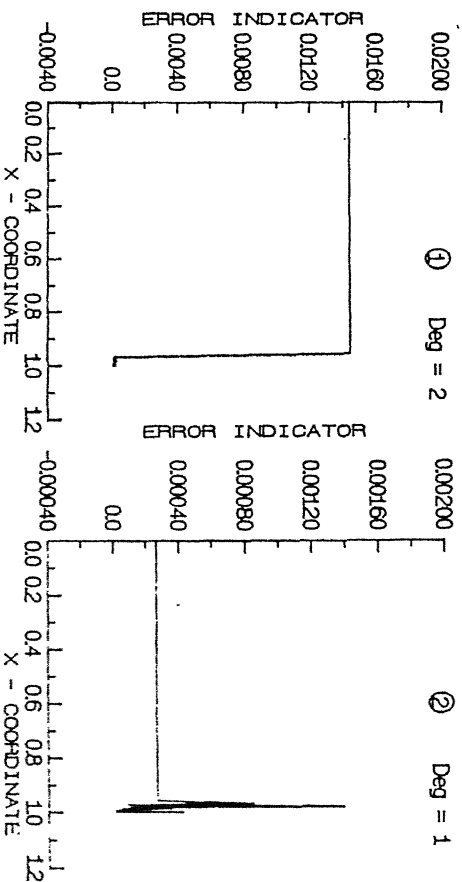


Graph ③

| Rdp | Nem | Do f |
|-----|-----|------|
| 3 | 43 | 57 |
| | | |

Graph ④

| Rdp | Nem | Do f |
|-----|-----|------|
| 4 | 43 | 59 |
| | | |



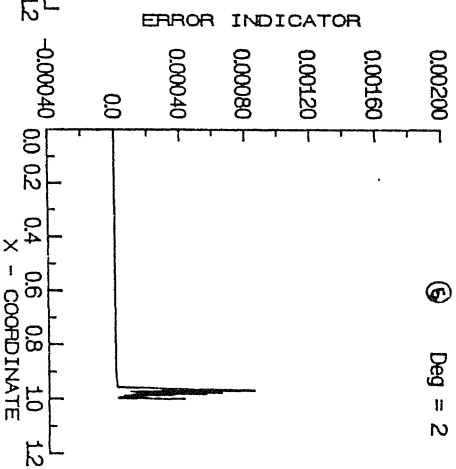
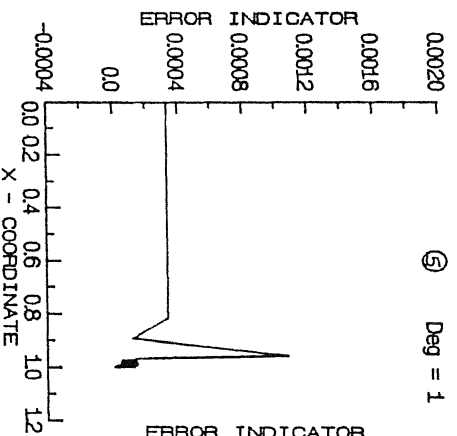
Graph ①

| Rdp | Nem | Do f |
|-----|-----|------|
| 1 | 43 | 44 |
| | | |

Graph ②

| Rdp | Nem | Do f |
|-----|-----|------|
| 2 | 43 | 56 |
| | | |

Fig. 4.28c. FLAME PROBLEM : BEHAVIOUR OF ERRORS (hp version)
(t = 0.00011)

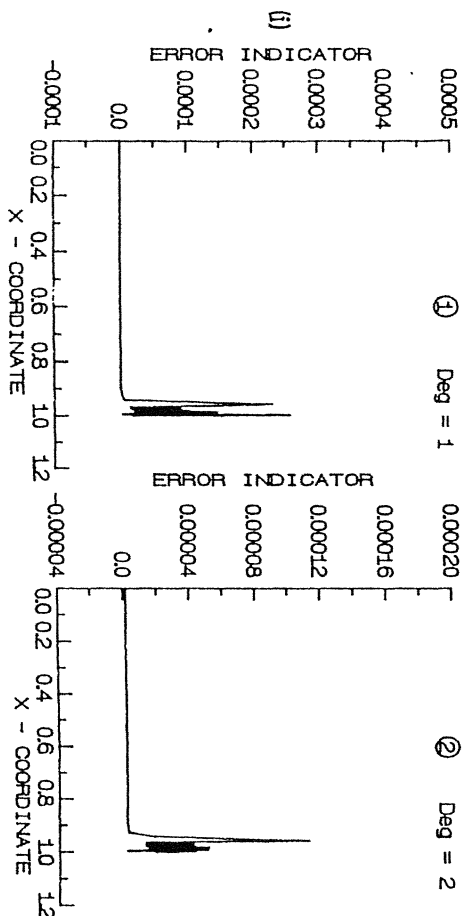


Graph ⑤

| Rdp | Nem | Dof |
|-----|-----|-----|
| 5 | 47 | 48 |

Graph ⑥

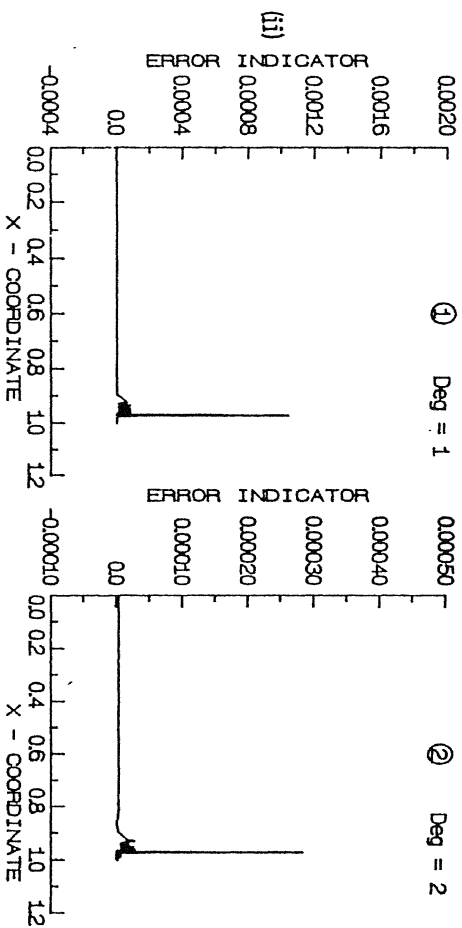
| Rdp | Nem | Dof |
|-----|-----|-----|
| 6 | 46 | 50 |



| Graph ① | | |
|---------|-----|-----|
| Pdb | Nem | DoF |
| 1 | 110 | 111 |

| Graph ② | | |
|---------|-----|-----|
| Pdb | Nem | DoF |
| 2 | 109 | 142 |

Fig. 4.28d FLAME PROBLEM : BEHAVIOUR OF ERRORS (hp VERSION)
 (i). $t = 0.000145$, (ii). $t = 0.0003$

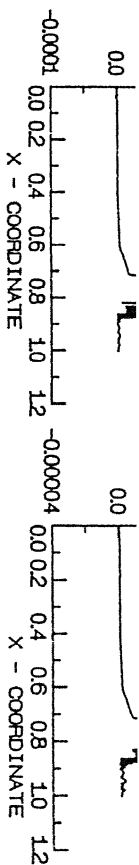


Graph ①

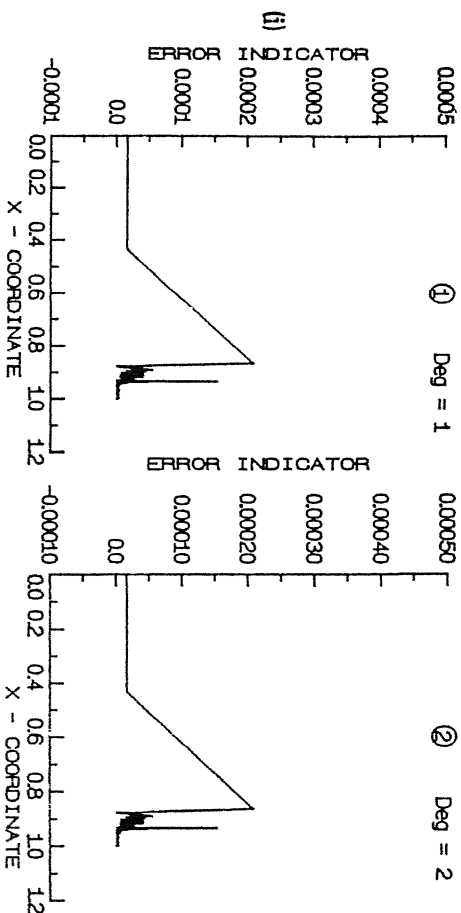
| PdP Name | DoF |
|----------|-----|
| 1 234 | 235 |
| | |

Graph ②

| PdP Name | DoF |
|----------|-----|
| 2 222 | 282 |
| | |



| | | |
|--|--|--|
| | | |
|--|--|--|



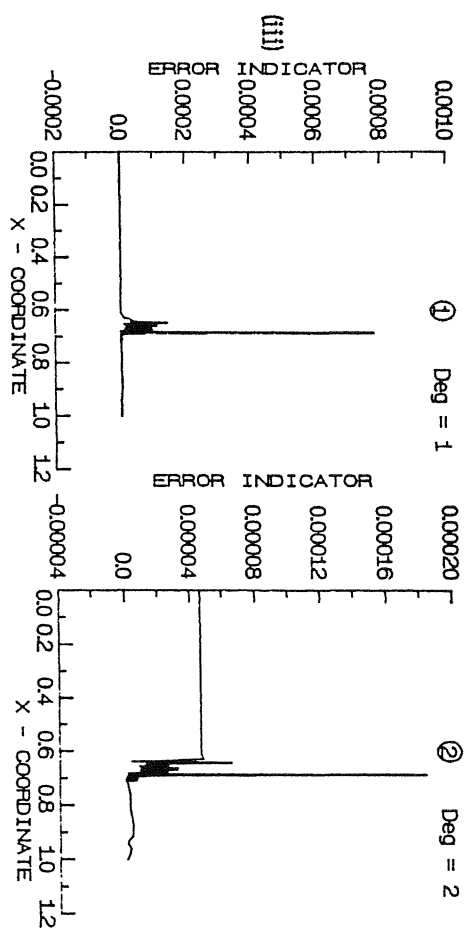
Graph (1)

| Rdp | Nem | Dof |
|-----|-----|-----|
| 1 | 239 | 248 |
| | | |

Graph (2)

| Rdp | Nem | Dof |
|-----|-----|-----|
| 2 | 229 | 238 |
| | | |

Fig. 4.28e. FLAME PROBLEM : BEHAVIOUR OF ERRORS (hp VERSION)
(i). $t = 0.000540$, (ii). $t = 0.001015$, (iii). $t = 0.002305$

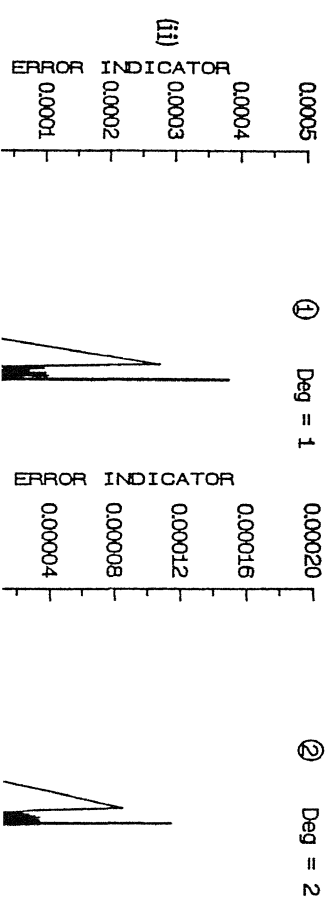


Graph ①

| Rdp | Nem | Do f |
|-----|-----|------|
| 1 | 194 | 195 |

Graph ②

| Rdp | Nem | Do f |
|-----|-----|------|
| 2 | 185 | 227 |



Graph ①

| Rdp | Nem | Do f |
|-----|-----|------|
| 1 | 206 | 207 |

Graph ②

| Rdp | Nem | Do f |
|-----|-----|------|
| 2 | 194 | 246 |

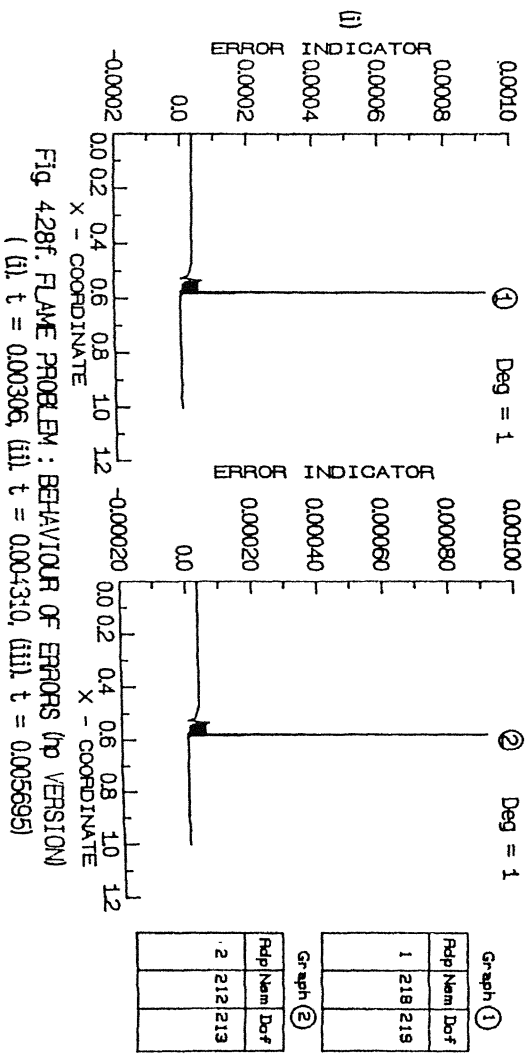
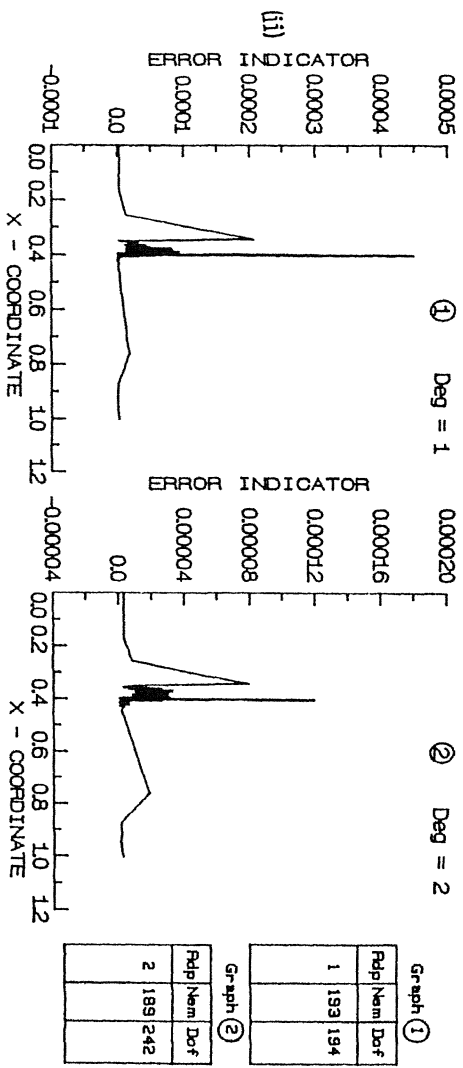
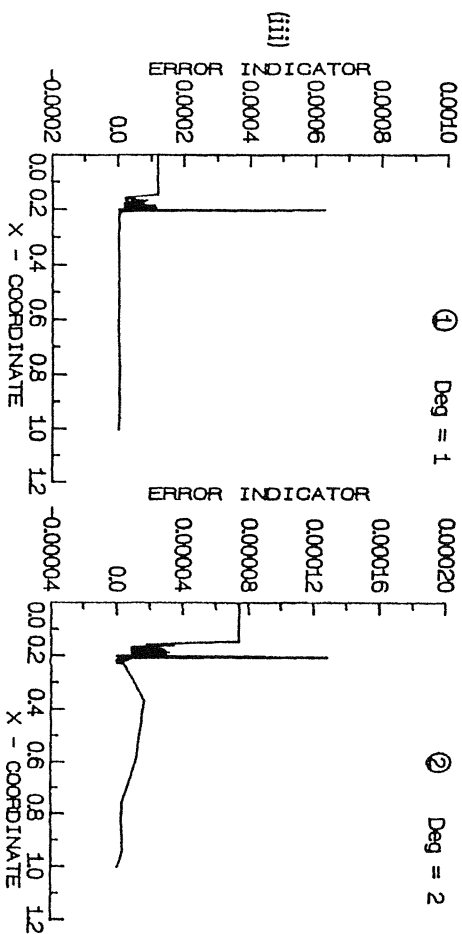


Fig. 4.28f. FLAME PROBLEM : BEHAVIOUR OF ERRORS (hp VERSION)
 (i). $t = 0.00306$, (ii). $t = 0.004310$, (iii). $t = 0.005695$



| Graph ① | | |
|---------|------|-----|
| Pdip | Norm | DoF |
| 1 | 189 | 198 |

| Graph ② | | |
|---------|------|-----|
| Pdip | Norm | DoF |
| 2 | 179 | 223 |

Figs. 4.29 a-c. The number of elements and the total degree of freedom in the problem have also been indicated for each instant of time and adaption level. Here, the grid adaption process has been divided into three steps, namely : (i) Derefinement where local error is below the minimum tolerance, (ii) Increase of interpolation degree in elements where the local error is above the mean value, (iii) conversion of p - version mesh into an approximately equivalent h mesh, for all the elements which have attained the maximum degree of interpolation permissible without meeting the tolerance criterion. For each time step, this procedure has been performed sequentially, until the global error criterion is met. During computation, each of these procedures have been carried out in a separate adaption step, for the sake of convenience. However, in the figures, only a few representative adaption steps have been shown for each time step and the numbers indicated do not correspond to the exact adaption step number.

It is observed from Figs. 4.28a and 4.29a that the local error decreases rather slowly when the degree of approximation is increased. Further, the decrease in the local error in some zones may lead to increase of the error in some neighbouring elements. Because of this, the number of adaptations required by hp method per time step is high. Additionally, the derefinement and the p - h conversion steps also increase the number of grid-adaptions. It is clear from these figures that the p refinement method by itself is not able to represent the steep profile of the local error distribution due to large element sizes. However, when p -version is coupled with h - version (by p - h conversion) the grid structure appears to have adequately small mesh sizes in

large error zones. It can also be seen from the table (4.3) that near the hump region, the p version results are not satisfactory and the local error indicator values have been polluted. But, after the conversion to h version and later applying the p version, the results in the hump region are substantially improved.

An important feature of mesh distributions in the h-p version (Figs. 4.29 a-c) is that for most of the time steps, the flame region requires not only linear degree elements but also higher degree elements. Also, near the flame boundaries (where sharp transitions occur in the solution profile) higher order elements are required. The local error indicator profiles in Figs. 4.28 a-f also reveal this feature in the form of wide peaks adjacent to the sharp peaks of the flame front region (see graphs for $t = 0.00003$, $t = 0.000145$, $t = 0.001015$ and $t = 0.00431$). The wide peaks arise due to large higher order elements; the narrow sharp peaks correspond to the highly refined linear elements.

The effectiveness of the p-h conversion is seen in the error indicator for the $t = 0.00011$ (Fig. 4.28c). The decrease of error with higher order interpolation is slow, when the number of elements remains same(p method). But when the p - version solution and mesh are converted into the equivalent h - version system, the fall in the error magnitude is rapid. The increase in the number of elements after conversion is insignificant. However, the refinement introduced by p-h conversion is non-uniform with closely spaced elements in the high gradient regions (see, Figs. 4.29a-b). Due to this, the h-p version

produces a well - graded mesh. The number of elements as well as the degree of freedom used in the h-p method are significantly lower than those required by the h-version, for maintaining the same global error criterion. It is worthy to note that the local error indicator values for the h-refinement procedure are normally smaller in magnitude than those of the p -refinement procedure. The reasons for this are the faster rate of convergence and the usage of larger number of elements in the h -method.

In the equilibrium phase of flame movement, the h-p method also converges fast (within 2 adaptions) for most of the time steps (see Figs. 4.28e-f and 4.29b-c). Features such as growth in the error band width during the ignition period (due to the growth in the front width), movement of the dense error band with respect to time (in order to cope up with flame movement) etc., are observed similar to the h-version graphs. The mesh distributions also have characteristics similar to those in the h-method. The flame region has dense mesh of linear elements while higher order elements occur in the region where the solution profiles have curved shape. In some situations, (say for $t = 0.00011$), when some elements in the non reactive zone become coarse due to derefinement, the degree of approximation increases near the flame front. In the successive time steps, these coarse higher order elements are refined properly into a non-uniform distribution of linear elements (by p to h conversion). In the flame front region, the introduction of new grid points after adaption have been restricted to only one in any element irrespective of the degree of approximation considered. This choice has been made in order to avoid over refinement in the

flame front-zone. In general, it has been observed that three steps of the h-p method, i. e. derefinement, increase of the interpolation order and p-h conversion, perform very well. In contrast to the h-version procedure, usually there is no movement of grid with adaption steps, in the h-p method. Only the degree of interpolation increases with adaption. Yet, the grid modification with time is excellent due to p-h conversion as the flame evolves through the ignition, equilibrium motion and extinction phases.

An overview of grid evolution is shown in Figs. 4.30a-b, for the h and hp versions respectively. Further, the results also have been shown in the tabular form (table 4.5 and 4.6a-b). The grid movement with respect to time is clearly seen to be uniform during the equilibrium phase. In a few places, the shift in the grid is not uniform due to unequal time intervals. Features such as the growth in flame front width during the ignition phase, constant front width during equilibrium movement etc. are observed, as discussed earlier. It is also evident that outside the flame region, the grid point density is slightly higher in the burnt region as compared to the unburnt region. This can be attributed to the fact that derefinement is relatively slower than refinement . At every location in the burnt region, the dense mesh has occurred once when the flame front passed through that location. It takes considerable time for derefining the dense mesh effectively. On the other hand, in the unburnt region, the mesh is always coarse due to the coarse initial mesh. Comparing the h and hp results, it is observed that the width of the dense mesh region is smaller for the hp region as compared to the h

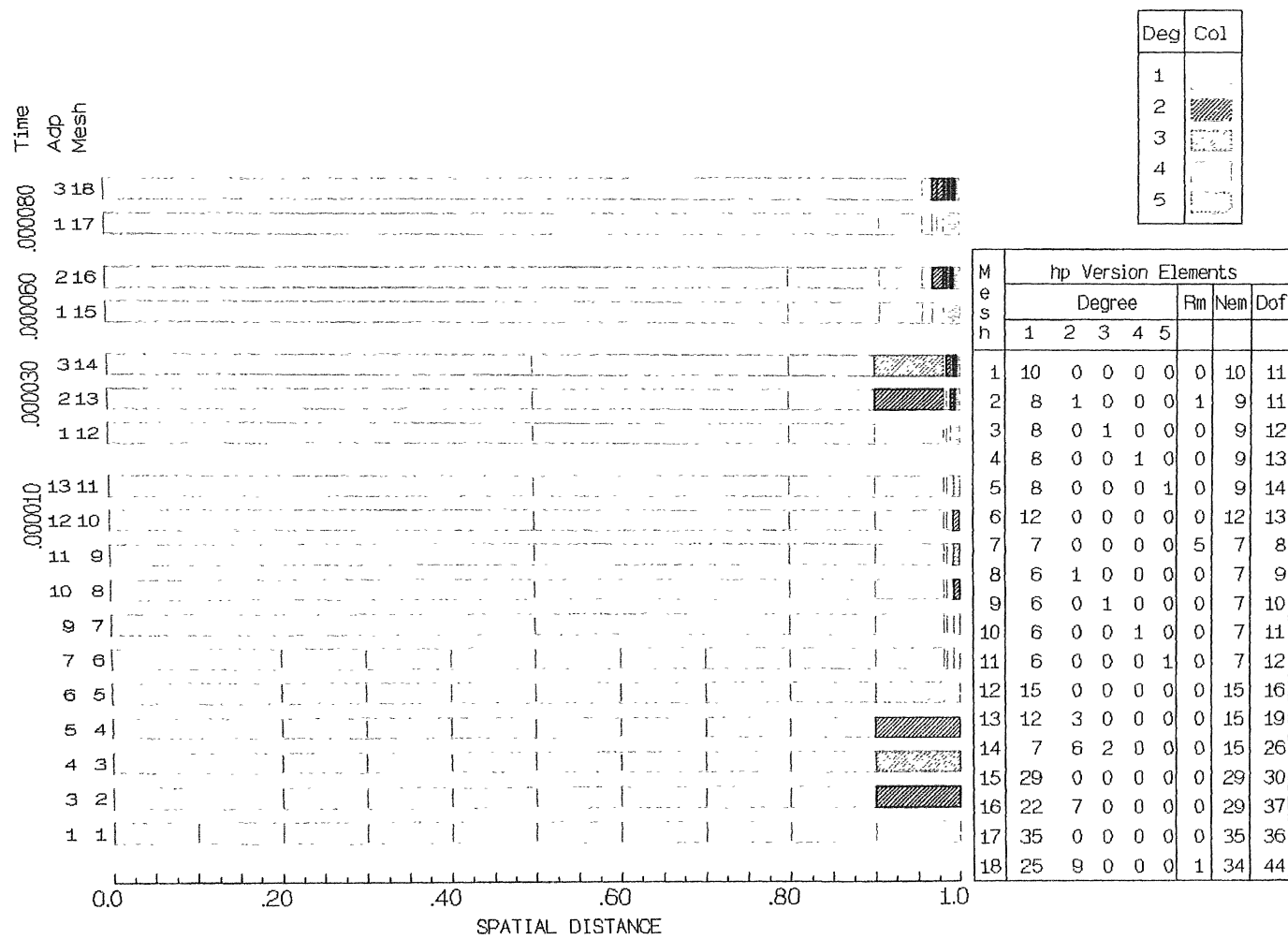


Fig. 4.29a hp - ADAPTED MESH AT VARIOUS TIME INSTANTS

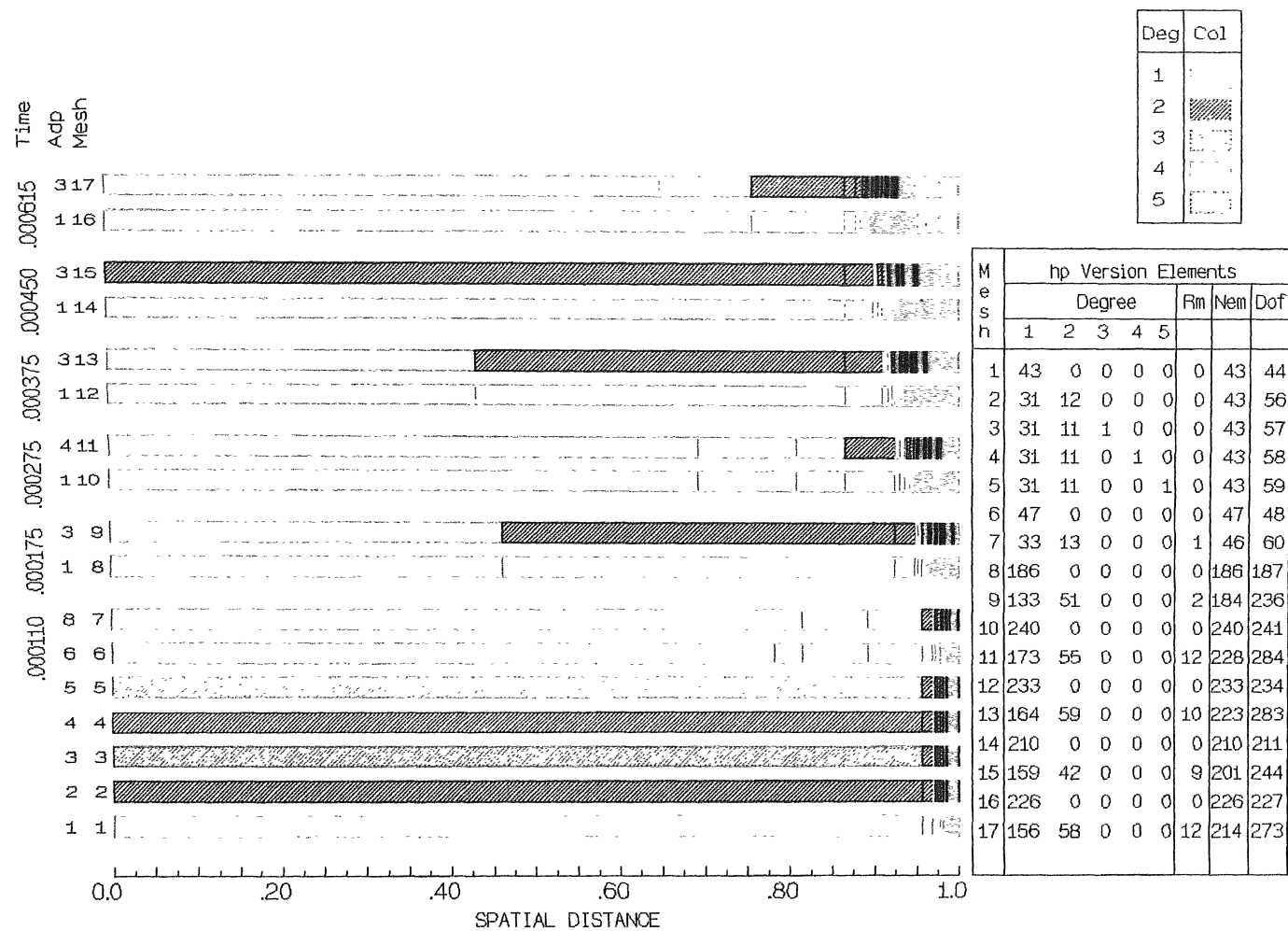
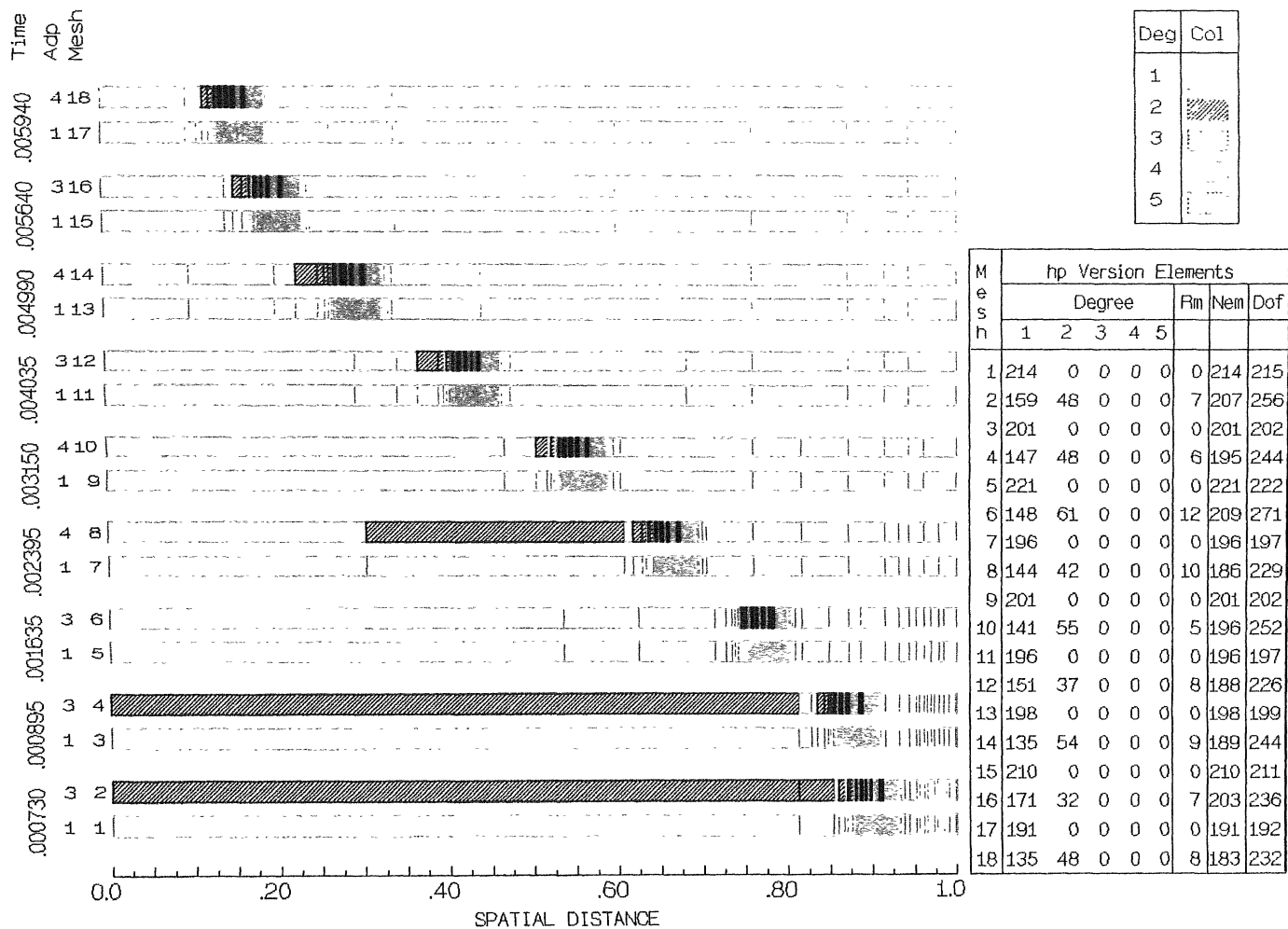
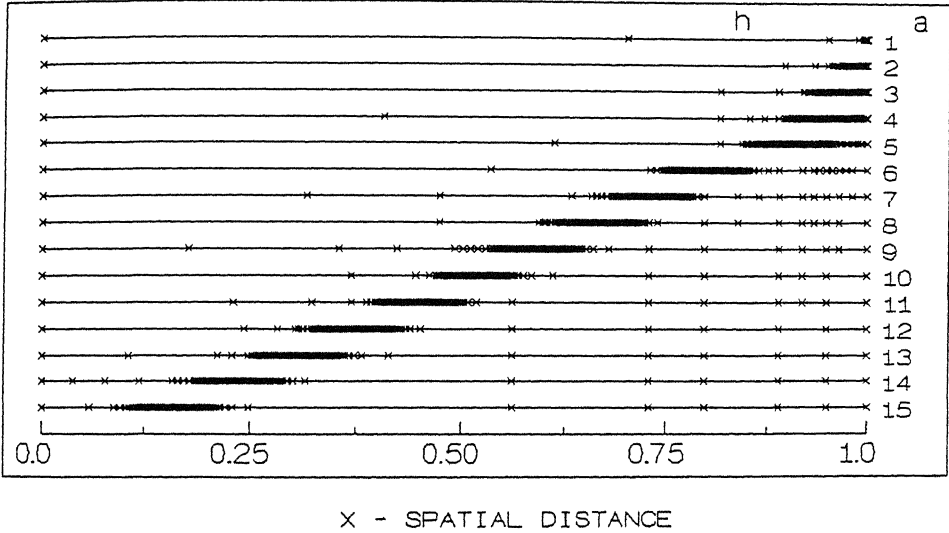
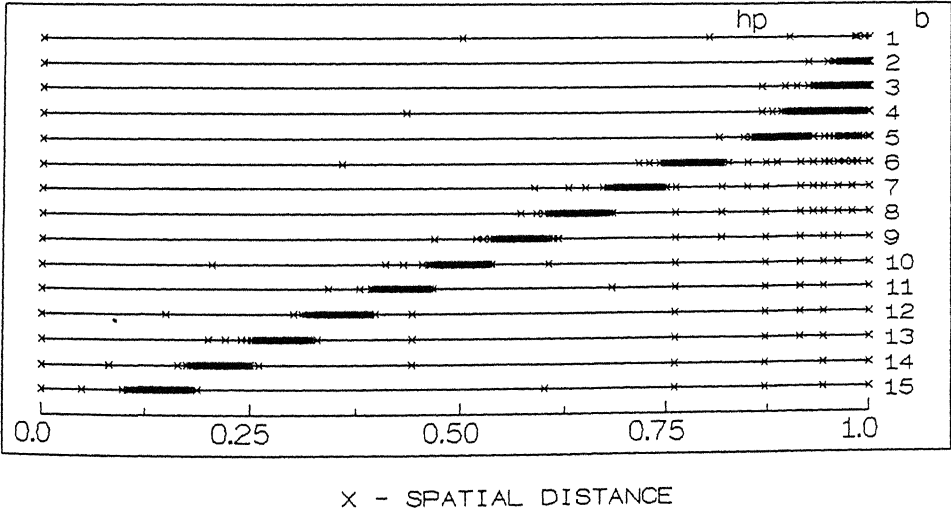


Fig. 4.29 b hp - ADAPTED MESH AT VARIOUS TIME INSTANTS





| Adp | Time | Eles | Nods |
|-----|----------|------|------|
| 1 | 0.000010 | 12 | 13 |
| 2 | 0.000150 | 141 | 142 |
| 3 | 0.000300 | 352 | 353 |
| 4 | 0.000550 | 370 | 371 |
| 5 | 0.000800 | 349 | 350 |
| 6 | 0.001550 | 361 | 362 |
| 7 | 0.002050 | 348 | 349 |
| 8 | 0.002550 | 369 | 370 |
| 9 | 0.003050 | 374 | 375 |
| 10 | 0.003550 | 329 | 330 |
| 11 | 0.004050 | 325 | 326 |
| 12 | 0.004550 | 399 | 400 |
| 13 | 0.005050 | 370 | 371 |
| 14 | 0.005550 | 343 | 344 |
| 15 | 0.006050 | 355 | 356 |



| Adp | Time | Eles | Nods |
|-----|----------|------|------|
| 1 | 0.000010 | 7 | 8 |
| 2 | 0.000150 | 139 | 140 |
| 3 | 0.000300 | 269 | 270 |
| 4 | 0.000550 | 211 | 212 |
| 5 | 0.000800 | 219 | 220 |
| 6 | 0.001550 | 246 | 247 |
| 7 | 0.002050 | 230 | 231 |
| 8 | 0.002550 | 207 | 208 |
| 9 | 0.003050 | 218 | 219 |
| 10 | 0.003550 | 217 | 218 |
| 11 | 0.004050 | 196 | 197 |
| 12 | 0.004550 | 229 | 230 |
| 13 | 0.005050 | 226 | 227 |
| 14 | 0.005550 | 221 | 222 |
| 15 | 0.006050 | 194 | 195 |

Fig.4.30 FLAME PROBLEM (h & hp version) ADAPTED MESH AT VARIOUS t

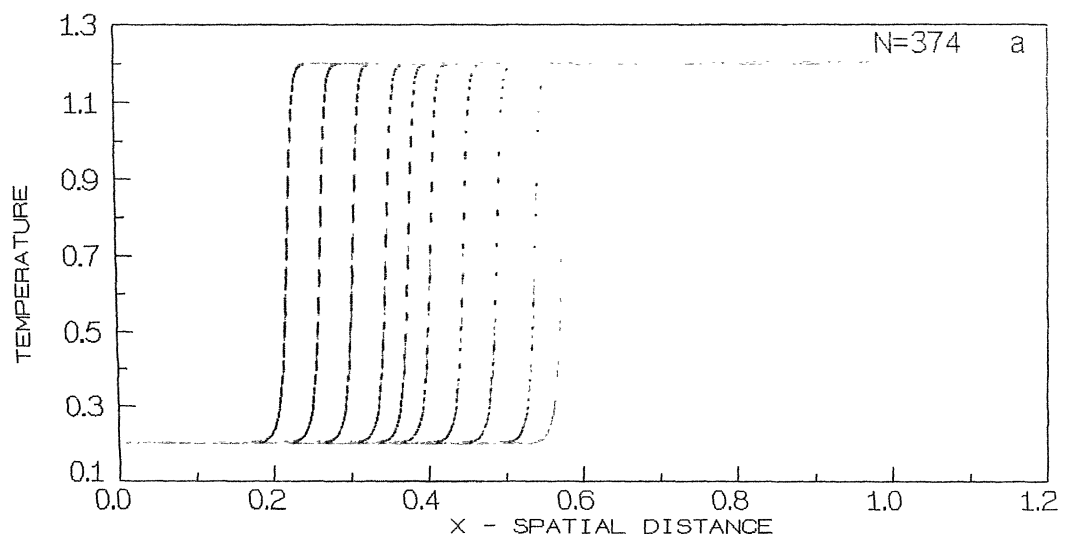


Fig. 4.31 FLAME PROBLEM (h version & Grid speed) : SOLUTION PROFILE

| Solution | Line_type |
|------------|-----------|
| h version | ----- |
| Grid speed | - - - - |

Fig. (a)

| Adp | Time | Eles | Nods |
|-----|-----------|------|------|
| 1 | 0 .003060 | 374 | 375 |
| 2 | 0 .003305 | 333 | 334 |
| 3 | 0 .003655 | 333 | 334 |
| 4 | 0 .003955 | 337 | 338 |
| 5 | 0 .004255 | 385 | 386 |
| 6 | 0 .004455 | 401 | 402 |
| 7 | 0 .004655 | 387 | 388 |
| 8 | 0 .004955 | 352 | 353 |
| 9 | 0 .005255 | 371 | 372 |
| 10 | 0 .005555 | 343 | 344 |

Fig. (b)

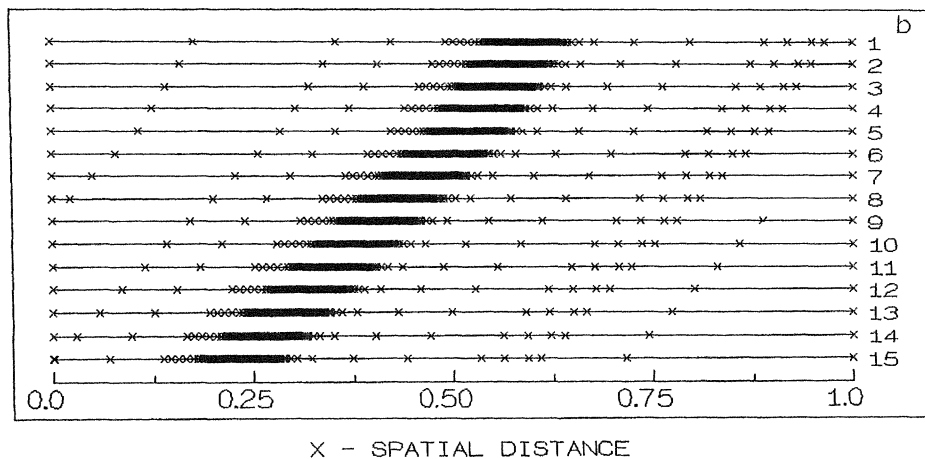


Fig. 4.31 FLAME PROBLEM (Grid speed) : ADAPTED MESH AT VARIOUS t

| Adp | Time | Eles | Nods |
|-----|-----------|------|------|
| 1 | 0 .003060 | 374 | 375 |
| 2 | 0 .003180 | 374 | 375 |
| 3 | 0 .003300 | 374 | 375 |
| 4 | 0 .003430 | 374 | 375 |
| 5 | 0 .003550 | 374 | 375 |
| 6 | 0 .003750 | 374 | 375 |
| 7 | 0 .003950 | 374 | 375 |
| 8 | 0 .004150 | 374 | 375 |
| 9 | 0 .004350 | 374 | 375 |
| 10 | 0 .004550 | 374 | 375 |
| 11 | 0 .004750 | 374 | 375 |
| 12 | 0 .004950 | 374 | 375 |
| 13 | 0 .005150 | 374 | 375 |
| 14 | 0 .005350 | 374 | 375 |
| 15 | 0 .005550 | 374 | 375 |

-version. This is explained from the fact that the edges of the front also require a fine mesh in h version, in order to represent the sharply changing temperature profile. On the other hand, for hp version, a few higher order elements can take care of sharp transition zones at the boundaries of the front.

(III). Results of the grid speed strategy :

The solution profiles (temperature) obtained by the grid speed strategy are compared with the h version results in Fig 4.31a. The problem starts with a non-uniform h version mesh of 374 elements at time $t = 0.003$. As discussed in the test problem results, the success of the grid speed strategy depends on the starting mesh at the initial time level. Therefore, the problem has to be solved by h or hp procedures until the flame front features are fully established and the corresponding solution and the mesh can be utilized for the application of the grid speed strategy for later times. In the figures., the shape and the movement of the fully established profiles for the h version and grid speed methods are compared over a range of time. It is observed that the agreement is quite satisfactory. After a sufficiently long application of the grid speed strategy (with an optimal h -version mesh in the beginning) collapse and stretching of elements occur near the boundary. This affects the behaviour of temperature as well as the density and therefore, special treatment of grid movement is done near the domain boundaries. As explained for the test problem, have also the nodes which cross the boundary are reintroduced at the beginning of the mesh, as can be seen from the Fig 4.31b. It is clear from this

figure that the nodes are distributed according to the front.

Thus, it can be concluded for the flame problem that the grid speed method provides an inexpensive and efficient way of constructing the grids. However, its applicability is limited to fronts of fixed speed and shape. The flame speed predicted by the present grid speed strategy agrees excellently with the results of Bieterman and Babuska (1986) who studied the same flame propagation problem by h version refinement strategy. A comparison between our results and those of Bieterman and Babuska (1982 a-b) indicates that the present results are satisfactory. The present results have also been shown in table (4.7).

The variation of the *a posteriori* error with time in the grid speed method is shown in the Fig 4.32b. Due to a good starting mesh and an accurate prediction of the flame speed, the *a posteriori* error remains more or less same with respect to time. In fact, it exhibits a small decrease in value, because of a slightly more optimal grid structure for the later times. Further, the maximum and mean errors with respect to time fluctuate only in the fourth decimal place and the mean of the errors with respect to time is almost constant. This study shows that the mesh constructed by the grid speed strategy is able to capture the front very efficiently.

4.8.3. Computational time :

The computational times for the h , h - p and grid speed algorithms have been presented in tabular form (tables 4.8, 4.9 and 4.10). The efficiency of these adaptive methods have been investigated and assessed in terms of the computer time required

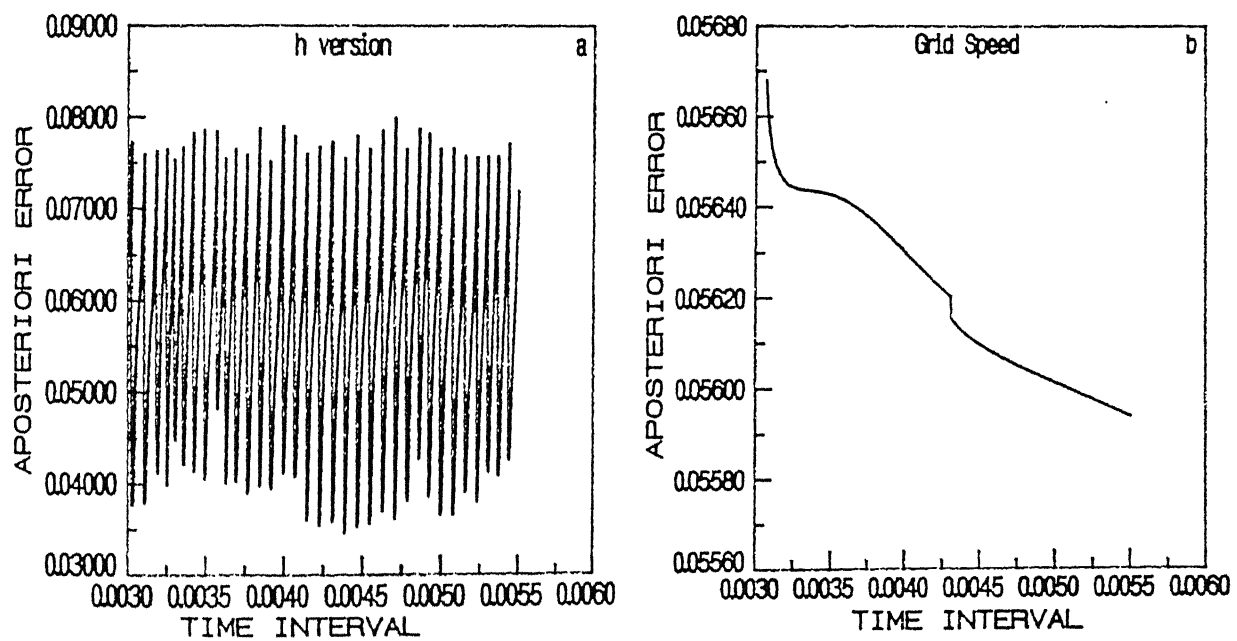


Fig. 4.32. FLAME PROBLEM : TIME EVOLUTION OF ERRORS (GRID SPEED)

| Time | N | Grid speed | Apos.err | Com.grd. |
|----------|-----|------------|-----------|----------|
| 0.003060 | 374 | -141.68312 | 0.0566779 | 8.761180 |
| 0.003160 | 374 | -142.07661 | 0.0564723 | 8.758090 |
| 0.003260 | 374 | -142.24686 | 0.0564417 | 8.754290 |
| 0.003360 | 374 | -142.34952 | 0.0564358 | 8.751800 |
| 0.003460 | 374 | -142.40625 | 0.0564302 | 8.750380 |
| 0.003560 | 374 | -142.43780 | 0.0564192 | 8.749590 |
| 0.003660 | 374 | -142.45676 | 0.0564008 | 8.749130 |
| 0.003760 | 374 | -142.46870 | 0.0563758 | 8.748850 |
| 0.003860 | 374 | -142.47665 | 0.0563462 | 8.748690 |
| 0.003960 | 374 | -142.48224 | 0.0563140 | 8.748590 |
| 0.004055 | 374 | -142.48623 | 0.0562824 | 8.748530 |
| 0.004160 | 374 | -142.48964 | 0.0562478 | 8.748490 |
| 0.004260 | 374 | -142.49226 | 0.0562159 | 8.748470 |
| 0.004360 | 374 | -142.49462 | 0.0561335 | 8.748460 |
| 0.004465 | 374 | -142.49656 | 0.0561042 | 8.748460 |
| 0.004560 | 374 | -142.49813 | 0.0560837 | 8.748470 |
| 0.004660 | 374 | -142.49959 | 0.0560652 | 8.748480 |
| 0.004760 | 374 | -142.50091 | 0.0560485 | 8.748490 |
| 0.004865 | 374 | -142.50216 | 0.0560322 | 8.748510 |
| 0.004965 | 374 | -142.50326 | 0.0560172 | 8.748520 |
| 0.005060 | 374 | -142.50424 | 0.0560032 | 8.748540 |
| 0.005170 | 374 | -142.50528 | 0.0559871 | 8.748560 |
| 0.005270 | 374 | -142.50616 | 0.0559723 | 8.748590 |
| 0.005370 | 374 | -142.50699 | 0.0559574 | 8.748610 |
| 0.005465 | 374 | -142.50771 | 0.0559432 | 8.748630 |
| 0.005555 | 374 | -142.50837 | 0.0559295 | 8.748650 |

Table. 4.7. Results of the grid speed strategy
for the flame propagation problem

| Dimension -less time | N | Nodes | Adp | Non-lin iter | CPU time/ time step (in seconds) |
|-------------------------|-----|-------|-----|-----------------|--|
| 0.000010 | 12 | 13 | 5 | 12 | 0.359625 |
| 0.000020 | 23 | 24 | 3 | 9 | 0.371920 |
| 0.000040 | 31 | 32 | 1 | 3 | 0.230394 |
| 0.000060 | 50 | 51 | 2 | 8 | 0.688824 |
| 0.000100 | 59 | 60 | 2 | 12 | 1.546934 |
| 0.000150 | 141 | 142 | 1 | 7 | 2.309051 |
| 0.000175 | 232 | 233 | 2 | 16 | 7.556550 |
| 0.000200 | 314 | 315 | 2 | 20 | 12.734232 |
| 0.000300 | 352 | 353 | 2 | 18 | 14.512045 |
| 0.001100 | 360 | 361 | 1 | 8 | 6.453213 |
| 0.001315 | 356 | 357 | 2 | 16 | 12.707541 |
| 0.002100 | 349 | 350 | 2 | 16 | 12.491621 |
| 0.002400 | 370 | 371 | 1 | 8 | 6.581406 |
| 0.003105 | 366 | 367 | 2 | 16 | 13.257330 |
| 0.003400 | 322 | 323 | 1 | 8 | 5.780185 |
| 0.004100 | 347 | 348 | 1 | 8 | 6.229663 |
| 0.004305 | 392 | 393 | 2 | 16 | 13.908195 |

Table. 4.8. CPU time of the h version algorithm
for the flame propagation

| Dimension -less time | N | Nodes | Non-lin iter | CPU time/ time step (in seconds) |
|-------------------------|-----|-------|-----------------|--|
| 0.003160 | 374 | 375 | 8 | 3.983766 |
| 0.003260 | 374 | 375 | 8 | 3.985091 |
| 0.003660 | 374 | 375 | 8 | 3.980361 |
| 0.004160 | 374 | 375 | 8 | 3.986336 |
| 0.004260 | 374 | 375 | 8 | 3.975732 |
| 0.004660 | 374 | 375 | 8 | 3.976706 |
| 0.005170 | 374 | 375 | 8 | 3.972914 |
| 0.005270 | 374 | 375 | 8 | 3.988380 |

Table. 4.9. CPU time of the grid speed strategy
for the flame propagation

| Dimension -less time | N | Dof | Adp | Non-lin iter | CPU time/ time step (in seconds) |
|-------------------------|-----|-----|-----|-----------------|--|
| 0.000020 | 10 | 16 | 2 | 5 | 0.157503 |
| 0.000040 | 15 | 26 | 2 | 9 | 0.391854 |
| 0.000060 | 29 | 37 | 2 | 6 | 0.452784 |
| 0.000100 | 43 | 44 | 1 | 6 | 0.579945 |
| 0.000150 | 139 | 140 | 1 | 7 | 2.967550 |
| 0.000175 | 184 | 236 | 2 | 11 | 8.570375 |
| 0.000200 | 235 | 236 | 1 | 9 | 10.794509 |
| 0.000300 | 222 | 282 | 2 | 13 | 13.575660 |
| 0.001200 | 227 | 228 | 1 | 9 | 8.142457 |
| 0.001295 | 219 | 220 | 1 | 9 | 7.697207 |
| 0.002000 | 238 | 239 | 1 | 9 | 8.772962 |
| 0.002100 | 234 | 235 | 1 | 9 | 8.778663 |
| 0.003060 | 218 | 219 | 1 | 9 | 7.018637 |
| 0.004005 | 191 | 235 | 2 | 11 | 9.046517 |
| 0.004300 | 200 | 201 | 1 | 8 | 6.442287 |
| 0.005095 | 199 | 252 | 2 | 11 | 10.145530 |

Table. 4.10. CPU time of the h-p version algorithm
for the flame propagation

to achieve a specified accuracy of the solution.

A computational code has been written in Fortran language and implemented on the CONVEX C-220 Machine for each of the adaption strategies. The CPU times presented here are obtained from a system call routine of the Vector library available on the CONVEX Machine. It is a well known fact that the CPU time and the storage requirements are parameters which are sensitive to the particular implementation of a numerical algorithm. In the present work, a high level of code optimization has not been performed and therefore, the conclusions which are drawn here may be considered to be only qualitative.

In the h-version algorithm, the CPU time has been calculated for both adaptive and non-adaptive time steps. Each of the important steps in the algorithm such as evaluation of elementary matrices, the calculation of elementary vectors during the inner iterations of the non-linear system of equations, decisions for the application of grid refinement, calculation of local and global error measures and the interpolation of old grid values to the new grid values, significantly contribute to the computational time. Thus, it is essential to look at the number of elements, number of adaptations within the particular time step and the total number of inner iterations in the non-linear solver, before interpreting the CPU time results. In table 4.8, it is seen that the CPU per time step is very small for initial times due to the small number of grid points. With grid refinement CPU increases in general, but occasional decrease in CPU also occurs if the number of adaptations and non-linear iterations to meet the convergence criteria are less for a particular time step(see, results for

$t = 0.00004$). A very sharp increase in CPU occurs near the completion of the ignition phase due to the high reaction rate around the hump region. Such an increase in CPU is contributed both by a dense refinement and a large number of adaptations and non-linear iterations. During the equilibrium phase of the flame propagation, the CPU times are more or less equal for the time steps without grid adaption (e.g. for $t = 0.0011$, 0.0024 , 0.0034 and $t = 0.0041$) and similarly for the time steps with grid adaption (e.g. for $t = 0.001315$, 0.0021 , 0.003105 and $t = 0.004305$). The minor differences in the calculated CPU have a good correlation with the number of grid points for each time level.

The CPU time results for the grid speed strategy are shown in Table 4.9. It is evident that this strategy is inexpensive compared to the other grid refinement algorithms. The CPU time is nearly constant for all time steps due to fixed number of grid points and non-linear iterations. The minor variations in CPU can be attributed to the special treatments applied for the boundary points.

For the hp version (see table 4.10), the trends are quite similar to those of h version. However, there are two important points to be noted here : (i) The hp version calculations need more CPU for the first adaption due to the additional evaluation of the hierarchical basis functions and the corresponding matrix entries, p -h conversion steps etc. (ii) For the later adaptations, the CPU time increase is much less due to the small number of additional calculations for the hierarchical degrees of freedom. Another attractive feature of the h-p refinement is that the

number of elements required to meet the specified tolerance is less.

It may be pointed here that numerical integration of elementary matrices is not necessary in the present work. However, in order to maintain the uniformity in all the adaptive methods, numerical integration has been adopted. The non-linear source term has been evaluated with eight point Gaussian quadrature formula. The assembled global matrix for linear degree basis functions is not stored for the h refinement calculations. Thus, optimization of the existing codes from the point of view of reducing storage memory and numerical computations is definitely possible and the CPU comparisons between these adaptive methods can then be made more rigorously with such optimized codes. However, the general features after optimization are also expected to be same as discussed in this section.

4.9. Conclusions :

In the present chapter, the h - and hp -methods have been described. These error based refinement algorithms have been applied to the test problem and the flame problem for a stationary mixture. The timewise variations of different error measures have also been analysed in detail. Apart from the h - and hp -methods, a dynamic adaptive strategy based on the grid velocity scheme has been applied. The computational results have been compared with analytical results for the test problem. For the flame problem a detailed comparative study of the h , h - p , and grid speed methods has been presented.

CHAPTER 5

A SELF ADAPTIVE ERROR INDEX BASED REFINEMENT STRATEGY
APPLIED TO FLAME PROPAGATION IN PRESENCE OF FLOW

5.1. Introduction :

This chapter describes a generalized h-refinement algorithm based on a self correcting index based a *posteriori* error estimate. The adaptive algorithm has been applied to the flame propagation and test problems corresponding to a flowing mixture. Numerical results of computable error measures have been presented for different values of the flow velocity. The solutions predicted by the h- version based α -strategy are compared with the solutions obtained by the standard h algorithm described in chapter 4.

5.2. Previous work on convective problems : Adaptive strategies :

Convective and diffusive effects occur frequently in association with heat and mass transfer processes in flow problems. Broadly speaking, the mathematical models of these processes contain both a " wave like " part due to convection and a smooth diffusive contribution. The manner in which these terms are treated numerically will influence the behaviour of the approximate solution. It is well known that the main difficulties arising in the numerical solution of the convective-diffusion equations are due to their non-self adjoint property and the mixed features of elliptic, parabolic and hyperbolic character of the governing equations. Galerkin method, which is well suited for

self-adjoint problems, leads to non-physical spatial oscillations when applied to highly convective situations, unless relatively fine meshes are employed. These oscillations are more pronounced in the vicinity of the front where the solution gradients are large. Similar difficulties have been noted during the usage of standard finite difference procedures also, which have paved the way for the development of upwinding difference schemes. Thus, the numerical method deserves special attention while developing reliable adaptive schemes to convective problems.

In order to eliminate the spurious spatial oscillations, various procedures have been employed over the years in finite difference methods, the most popular being the 'upwind' differencing of the convective term. A number of numerical studies by finite difference methods have been reported in the recent past on such problems. The details of these studies have been summarized by Raithby, 1976; Leonard, 1981; Patel et al., 1985). Also, several authors (Thompson, 1984; Acharya and Patankar, 1985, 1990; Ramos, 1985; Chargy et al., 1990; Eriksson and Johnson, 1990; Rai and Anderson, 1982, Dwyer et al., 1980; Demkowicz and Oden, 1986) have proposed various adaptive schemes to solve convective diffusion problems. Most of these adaptive methods attempt to equi-distribute some measure of the error, but differ in the individual approaches.

Among the various techniques employed to deal with convective diffusion problem, the construction of upwind finite element approximation via the Galerkin method with added artificial diffusion is an important one (Heinrich et al., 1977; Hughes, 1978; Christie and Mitchell, 1978; Barrett et al., 1980;

Szymczak and Babuska, 1984; Babuska and Rheinboldt, 1980, 1982; Zienkiewicz and Heinrich, 1978; Carey and Oden, 1984b). When upwinding is applied in finite element methods, modified weighted basis functions are employed to achieve the desired effect. Here, the test functions belong to a space different from that of standard shape functions. In the upwinding strategy, the criterion for the selection of optimal upwinding factor w has been either the elimination of oscillations or the attainment of exact nodal solutions for certain class of model problems. The concepts that the degree of upwinding can be continuously controlled to increase the accuracy of the numerical solution and that the oscillations can be removed by several mesh refinements, has been widely used in the references cited above.

The theory developed by Babuska and Rheinboldt (1978a,b, 1982) has also been widely used in the development of adaptive schemes for convective diffusive problems. These adaptive procedures are based on the *a posteriori* error estimates for a norm which arises from symmetrizing the bilinear form. The method of symmetrization has been studied by Barret and Mortan (1980) and has been used in the optimal FE solutions to convective diffusive problems. Later, Szymczak and Babuska (1984) have derived the *a posteriori* error estimates which are asymptotically exact for convective diffusive problems with upwinding finite elements. Further, the authors also suggested ways of optimising the upwind parameter w for certain class of problems. This *a posteriori* error estimate depends on local mesh size h and the local elemental error indicators calculated from the residual of the governing equations. The effectiveness of such an error estimate

has been studied by measuring the effectivity index. A variable local upwinding strategy has been employed in conjunction with the adaptive mesh refinement by the authors Carey and Oden (1984b).

Previous adaptive techniques (Johnson, 1989; Lohner et al., 1986) for convective-diffusive problems, have been based either on a priori interpolation error estimates for refining the mesh locally according to the size of the gradient or on a *posteriori* error estimates which depend only on the residue. For both the approaches the reliability is uncertain. In the first approach, the derivatives of the exact solution are used without considering the computed solution and in the second case, the residue may be large in regions of non-smoothness. Eriksson and Johnson, (1990) have developed adaptive streamline diffusion FE methods for time dependent convective diffusive problems. These adaptive methods employ a *posteriori* error estimates and the concept of space-time elements. The developments of Streamline Upwind Petrov-Galerkin method (SUPG) and the Characteristic Galerkin method (CG) have led to different approaches for solving convection dominated problems (Brooks and Hughes, 1982; and Zienkiewicz et al., 1984; Douglas and Russell, 1982; Oden and Demkowicz, 1986b).

5.3. Description of the present work :

In the past, a *posteriori* spatial error estimators have been developed for self-adjoint and non-self adjoint linear operators based on the error analysis for the finite dimensional approximate solution. However, such error analysis is generally available for linear operators. On a tentative basis, the same error estimates for linear operators have been used for non-linear problems also

in the past. Such usage, however is clearly ad hoc.

In the present work, a generalized error estimator is proposed which can be applied to linear or non-linear problems. It is only assumed that *a posteriori* error depends on the step size raised to unknown power α and the value of α is determined in the course of the adaption process itself. In this sense, the method has the feature of providing a self correcting error estimator. The strategy is based on computational study only and no mathematical theory is developed. The refinement strategy has been applied to the test problem and the flame propagation problem in the presence of mixture flow.

In this section, we describe the test problem and the flame problem with convection. The details of the upwind finite element solution for these problems are also outlined.

5.3.1. Description of Test problem :

The test problem formulated here is an extension of the test problem described in Chapter 4 (see eqs. 4.1a-c) .. The first order convective term is introduced and the modified governing equation for the test problem can be written as

$$\frac{\partial}{\partial t} u(x,t) - \frac{\partial}{\partial x} \left(a(x) \frac{\partial}{\partial x} u(x,t) \right) + v \frac{\partial}{\partial x} u(x,t) + b(x) u(x,t) = \bar{f}(x,t) \\ (x,t) \in I \times (0, t_f) \quad (5.1)$$

The boundary and initial conditions are same as described earlier. The source term $\bar{f}(x,t)$ has been chosen appropriately so that the exact solution of the equation (5.1.) has similar characteristics in the wave front region for all velocities v . For computational purpose, we have selected

$$a(x) = 1, \quad b(x) = 0 \quad \forall x \in (0,1)$$

The exact solution of the eq. (5.1) is given by

$$U(x,t) = 1/2 + 1/2 \tanh [2\beta(x-Ct)]$$

with same values of β and C as described in the Chapter 4. (see eq. (4.2)).

5.3.2. Description of Flame problem in presence of flow :

The governing equations of the flame propagation problem in the presence of flow can be written as follows :

$$\left. \begin{aligned} \frac{\partial T}{\partial t} - \frac{\partial^2 T}{\partial x^2} + v \frac{\partial T}{\partial x} &= f(\rho, T) \\ \frac{\partial \rho}{\partial t} - \frac{\partial^2 \rho}{\partial x^2} + v \frac{\partial \rho}{\partial x} &= -f(\rho, T) \end{aligned} \right\} \quad t > 0, \quad x \in (0,1) \quad (5.2)$$

where v is the prescribed fluid velocity. The boundary and initial conditions for this problem have already been defined in equations (3.1b-f) of Chapter 3. The above equations can be expressed in the more general form as follows :

$$(u_1)_t - \left(a_1(x) (u_1)_x \right)_x + v (u_1)_x = f_1(x, t, u); \quad \begin{matrix} (x, t) \in \Omega \times (0, t_f] \\ 1 = 1, \text{ NPDE} \end{matrix} \quad (5.3)$$

During the computations, $a_1(x)$ has been set equal to unity in all the equations, for the sake of simplicity.

5.3.3. Basic formulation:

This section describes the mathematical frame work in which the above test problem (eq. 5.1) and flame problem (eq. 5.2) will be studied. The notation used and the definitions of the variables employed here, are analogous to those defined in Chapters 3 and 4.

Semi discrete approximation : The finite element formulation and the method of solution have been explained in detail Chapter 3. Here, only the important steps in the FE formulation have been discussed. Following precisely the same procedures employed in Chapter 3, the semi discrete approximation for the eqs. (5.3) can be written as :

$$\left\langle \frac{\partial}{\partial t} U_1(\cdot, t), V_1 \right\rangle + \left\langle a_1 \frac{\partial}{\partial x} U_1(x, t), \frac{\partial}{\partial x} V_1 \right\rangle + \left\langle \nu \frac{\partial}{\partial x} U_1(x, t), V_1 \right\rangle \\ = \left\langle f_1(x, t, U), V_1 \right\rangle$$

$$\text{for all } U_1, V_1 \in S_h^{r,k}; \quad t \in (0, t_f], \text{ and } i = 1 \dots, NPDE. \quad (5.4a)$$

with suitably chosen initial conditions. Expressing $U_1(x, t)$ in terms of basis, the above system of equations can also be written in a matrix form as

$$[M_1] \left\{ \frac{dU_1[t]}{dt} \right\} + \left[[K_1^1] + [K_1^2] \right] \{U_1[t]\} = \{F_1(U[t])\} \\ \text{for } i = 1, \dots, NPDE \quad (5.4b)$$

where

$$[K_1^1] = [(K_1^1)_{ij}] \text{ and } [K_1^2] = [(K_1^2)_{ij}]; \quad (1 \leq i \leq N_h), \quad (1 \leq j \leq N_h)$$

are the matrices with entries

$$(K_1^1)_{ij} = \langle a_1 \nabla \psi_{1i}, \nabla \psi_{1j} \rangle$$

and

$$(K_1^2)_{ij} = \langle a_1 \nabla \psi_{1i}, \psi_{1j} \rangle.$$

Here, N_h is the dimension of the finite dimensional space $S_h^{r,k}$ and $U_1[t]$ ($1 \leq i \leq NPDE$) is the vector of nodal degrees of freedom which is determined as the solution of the system of time dependent differential equations (5.4b). The other terms in

the above equation are defined in the same fashion as in Chapter 3.

upwinding formulation :

In the present study, the concept of upwinding finite element technique has been employed in the solution procedure. A linear combination of special test functions (upwinding basis functions) with piecewise linear basis functions have been used for test space of solutions in the weak formulation (see Szymczak and Babuska, 1984). This is achieved by adding a quadratic term, multiplied by some parameter w to each linear basis function of the test space. The construction of upwind finite basis functions for a particular element is explained below :

Let $\Psi_{1,1}(\xi)$ and $\Psi_{1,2}(\xi)$; $-1 \leq \xi \leq 1$; $i = 1, \text{ NPDE}$, be the usual piece-wise linear basis functions on the master element $\hat{\Omega}_n = [-1,1]$. Using standard transformation procedures, these local basis functions can be used to obtain the global basis functions $\psi_{1,1}(x)$ and $\psi_{1,2}(x)$. The upwind weighted element functions (test functions for the element $\hat{\Omega}_n$) can be written as follows.

$$\begin{aligned}\hat{\Psi}_{1,1}(\xi) &= \Psi_{1,1}(\xi) + w \Psi_{1,1}(\xi) \Psi_{1,2}(\xi) \\ \hat{\Psi}_{1,2}(\xi) &= \Psi_{1,2}(\xi) - w \Psi_{1,1}(\xi) \Psi_{1,2}(\xi)\end{aligned}\tag{5.5}$$

where w is a scaling factor that specifies the amount of upwind bias desired. Here, the product $\Psi_{1,1}(\xi) \Psi_{1,2}(\xi)$ is the additional quadratic form in the test function and it vanishes at the end points of $\hat{\Omega}_n$ (Fig. 5.1). Further, the sign in the expression indicates the upwind direction which may vary according to the

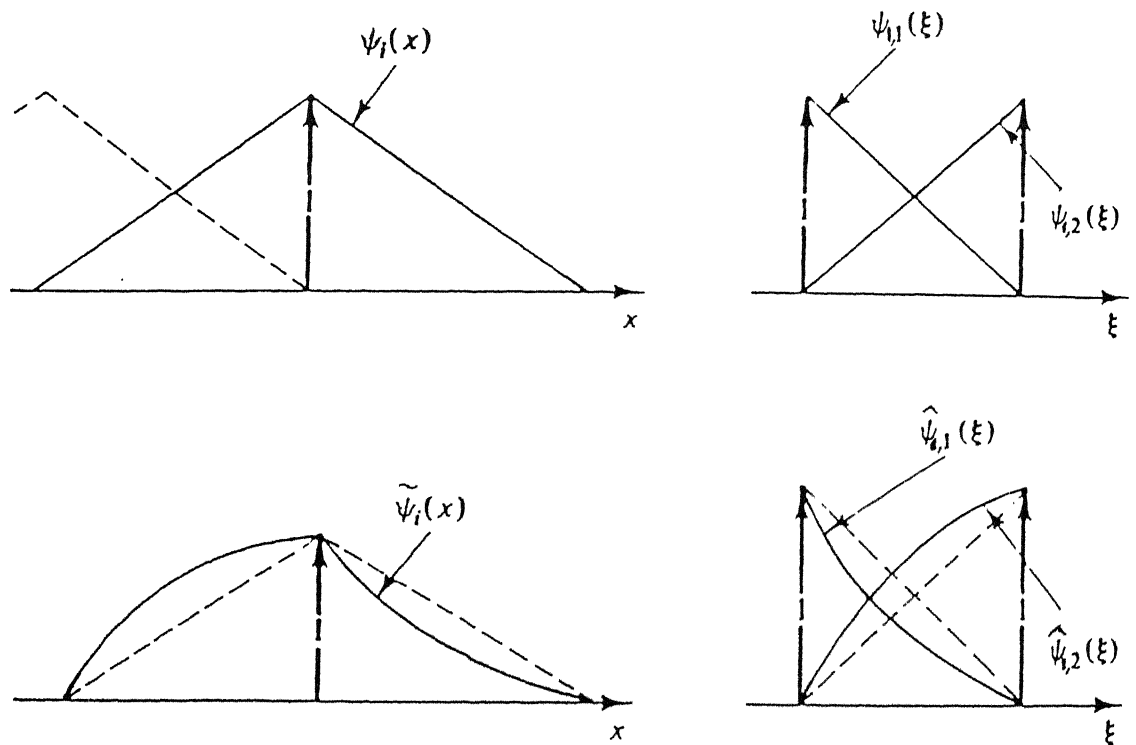


Fig. 5.1. Construction of quadratically biased upwind test functions.

problem. For the flame problem in which the front propagates from right to left, the upwind direction is defined from right to the left. Thus, at every node, a quadratic bias is added or subtracted from the standard linear basis function corresponding to the region ahead or behind the node respectively, in the direction of propagation. Using the upwind biased basis functions ($\hat{\Psi}_{1,1}(\xi)$, $\hat{\Psi}_{1,2}(\xi)$) as test functions for the master element $\hat{\Omega}_n$, one can construct the global test functions with the desired properties (see Fig. 5.1.).

Representation of the solution space :

In order to obtain a finite dimensional approximate solution of equation(5.3), the finite dimensional solution spaces for the test and trial functions must be specified. For the sake of convenience, we recall the definition of $S_h^{r,k}(\Delta)$ which has been used in the Chapter 4.

Trial space : We denote $S_h^{r,k}(\Delta)$ as the trial space which consists of all $(r-1)$ times continually differentiable functions on Ω_j for which the restriction to Ω_j , $j = 0, 1, 2, \dots, N_h - 1$ is a polynomial of degree $k_j - 1$ and $k_j \geq 2r$, $j = 0, 1, 2, \dots, N_h - 1$.

Test space : The test space $S_h^{r,\tilde{k}}(\Delta)$ is spanned by two sets of basis functions, namely, the linear basis functions

$$\psi_1 = \left\{ \psi_{1,j} \right\} ; j = 1, 2, \dots, N_h$$

and the upwinding basis functions

$$\tilde{\Psi}_1 = \left\{ \tilde{\Psi}_{1,j} \right\} \quad j = 1, 2, \dots, N_h$$

where, the upwinding basis functions $\tilde{\Psi}_{1,j}(x)$ are constructed by

the procedure described above. It is to be noted that the vector $\tilde{\mathbf{k}}$ in the test space and \mathbf{k} in the trial space are not equal, as their components may differ from each other due to different degrees of approximation considered locally in each element of the solution domain. In general, the basis functions in the test space are denoted by

$$\alpha_1 = \left\{ \alpha_{1,j}(x) \right\} ; j = 1, 2, \dots, N_h$$

where $\alpha_{1,j}(x)$ is a linear combination of

$$\psi_{1,j}(x) \text{ and } \tilde{\psi}_{1,j}(x)$$

(5.6)

Implementation of Upwinding :

To generate the desired upwind model, the test space $S_h^{r,\tilde{\mathbf{k}}}(\Delta)$ described above is considered for the purpose of constructing the approximation. Using the following test functions,

$$V_1 = \alpha_1 ; \alpha_1 \in S_h^{r,\tilde{\mathbf{k}}}(\Delta), \text{ for } 1 = 1, \dots, \text{NPDE}$$

the semi discrete equations in the upwind formulation can be written as :

$$\begin{aligned} \left\langle \frac{\partial}{\partial t} U_1(\cdot, t), \alpha_1 \right\rangle + \left\langle a_1 \frac{\partial}{\partial x} U_1(x, t), \frac{\partial}{\partial x} \alpha_1 \right\rangle + \left\langle v \frac{\partial}{\partial x} U_1(x, t), \alpha_1 \right\rangle \\ = \left\langle \tilde{f}_1(x, t, U), \alpha_1 \right\rangle \end{aligned}$$

$$\text{for all } U_1 \in S_h^{r,\mathbf{k}}(\Delta); \quad 1 = 1, \dots, \text{NPDE}.$$

(5.7a)

with suitably chosen initial conditions. The resulting equations (5.7a) can be written in matrix form as :

$$\begin{aligned}
& \left[\begin{bmatrix} M_1 \end{bmatrix} + w \begin{bmatrix} \tilde{M}_1 \end{bmatrix} \right] \left\{ \frac{dU_1[t]}{dt} \right\} + \left[\begin{bmatrix} K_1^1 \end{bmatrix} + \begin{bmatrix} K_1^2 \end{bmatrix} + w \begin{bmatrix} \tilde{K}_1^1 \end{bmatrix} + w \begin{bmatrix} \tilde{K}_1^2 \end{bmatrix} \right] \left\{ U_1[t] \right\} \\
& = \left\{ F_1(U[t]) \right\} + w \left\{ \tilde{F}_1(U[t]) \right\} \\
& \text{for } 1 = 1, 2, \dots, NPDE
\end{aligned} \tag{5.7b}$$

In the above equation, the quantities with '~' have been contributed by the product terms of the upwind basis functions while the other matrix contributions are due to standard basis functions in the test space. Also, the superscripts 1 and 2 indicate diffusive and convective contributions respectively.

The upwind matrices are given by

$$\begin{aligned}
\begin{bmatrix} \tilde{K}_1^1 \end{bmatrix} &= [(\tilde{K}_1^1)_{ij}] \text{ and } \begin{bmatrix} \tilde{K}_1^2 \end{bmatrix} = [(\tilde{K}_1^2)_{ij}] \\
(1 \leq i \leq N_h), & (1 \leq j \leq N_h)
\end{aligned}$$

with entries

$$\begin{aligned}
(\tilde{K}_1^1)_{ij} &= \langle a_1 \nabla \psi_{1i}, \nabla \alpha_{1j} \rangle \\
(\tilde{K}_1^2)_{ij} &= \langle a_1 \nabla \psi_{1i}, \alpha_{1j} \rangle.
\end{aligned}$$

and the upwind vector $\left\{ \tilde{F}_1(U[t]) \right\}$ has entries

$$\tilde{F}_{1j} = \langle \tilde{f}_1(x, t, U[t], \alpha_j) \rangle.$$

5.4. Description of present a posteriori error estimate :

A major contribution of the present work relates to the development of a generalized equi-distribution strategy (designated as the α -strategy) using for local error indicator and the *a posteriori* error. The error is estimated in terms of the residual of the finite element solution and the local mesh size parameter raised to an unknown index α . The only assumption

invoked for the application of the α -strategy is that the error estimator depends on the product of a measure of the local residue in some properly chosen norm and a positive power of the local mesh size. Based on these requirements, the *a posteriori* error estimate $\varepsilon^{1,m}(t,\alpha)$ at $t = t_1$ for the m^{th} adaption can be written as follows :

$$\varepsilon^{1,m}(t,\alpha) = \left(\sum_{n=1}^{N_e^{1,m}} E_n^{1,m}(t,\alpha) \right)^{1/2} = \left(\sum_{n=1}^{N_e^{1,m}} \sum_{l=1}^{NPDE} |\eta_{l,n}^{1,m}(t,\alpha)|^2 \right)^{1/2} \quad (5.8a)$$

where the error indicator $\eta_{l,n}^{1,m}(t)$ is given by

$$|\eta_{l,n}^{1,m}(t,\alpha)|^2 = h_n^\alpha \int_{\Omega_n^{1,m}} (r_{l,n}^{1,m}(t,s))^2 ds \quad (5.8b)$$

Here, $r_{l,n}^{1,m}(t,x)$ is the residue of the finite element solution in n^{th} element which is computable for each $t > 0$ on the sub interval (x_n, x_{n+1}) .

5.5. Scope of the present study :

The aim is to develop a computational procedure which meets the following goals :

- (G1). To develop a versatile, error based refinement strategy which has the capability of producing sufficiently accurate numerical solutions of non-linear parabolic partial differential equations. The accuracy is to be measured in some properly chosen norm associated with the problem.
- (G2). To develop FE solutions for the flame and test problems in the presence of flow which satisfy a prescribed *a posteriori*

error criterion.

- (G3). To obtain the proper distribution of grid points for h-version α strategy at each instant of time for the physical problems stated in (G2).
- (G4). To present a detailed study of behaviour of various error measures for this new algorithm and compare the relative performance of the proposed error estimate with that of standard h refinement strategy.

5.5.1. Mathematical statement :

In this subsection, the objectives are expressed in the mathematical frame work. Firstly, the global residue norm $\mathcal{R}^{i,m}(t)$ is expressed in terms of elemental residue norms $\mathcal{R}_n^{i,m}(t)$ as

$$\left(\mathcal{R}^{i,m}(t) \right)^2 = \sum_{n=1}^{\mathcal{N}_e^{i,m}} \mathcal{R}_n^{i,m}(t) = \sum_{l=1}^{\mathcal{N}_e^{i,m}} \sum_{n=1}^{NPDE} \int_{\Omega_n^{i,m}} r_{l,n}^{i,m}(s,t)^2 ds \quad (5.9a)$$

We define the relative global residue $\mathcal{R}_{REL}^{i,m}(t)$ for m^{th} ($m > 1$) adaption at time $t = t_i$ as

$$\mathcal{R}_{REL}^{i,m}(t) = \left(\mathcal{R}^{i,m}(t) - \mathcal{R}^{i,m-1}(t) \right) / \mathcal{R}^{i,m}(t) \quad (5.9b)$$

The immediate goal is achieved in two steps. Given a problem (P) with computed solution $\bar{U}(x,t)$ and certain measure of error the tasks to be completed are :

- (a). For the initial time interval $(t_0, t_1]$, design an adaptive algorithm based on equi-distribution of local error indicator for constructing a finite element mesh ' Δ ' such

that the residues $\mathcal{R}^{i,m-1}(t)$ and $\mathcal{R}^{i,m}(t)$ for two successive adaptions $(m-1, m)$ satisfy the relation

$$\mathcal{R}_{REL}^{i,m}(t) \leq \delta_{REL} \quad (5.10a)$$

and determine the index α by setting

$$\mathcal{E}^{i,m}(t, \alpha) - \delta_1 = 0 \quad (5.10b)$$

where $\mathcal{E}^{i,m}(t, \alpha)$ is defined in the equation (5.8b). Further, the equi-distribution is performed with the help of the statistical procedure.

- (b). For the subsequent time intervals, the task is to design an adaptive algorithm for constructing the finite element mesh ' Δ ' such that *a posteriori* error $\mathcal{E}^{i,m}(t, \alpha)$ satisfies the relation

$$\delta_1 \leq \mathcal{E}^{i,m}(t, \alpha) \leq \delta_2 \quad \forall t \in (t_1, t_f) \quad (5.11a)$$

where α is known apriorily or determined from the relation

$$\mathcal{E}^{i,m}(t, \alpha) - \delta_1 = 0 \quad (5.11b)$$

The decisions concerning mesh modification are aimed at achieving (5.11a) at different time instants. Whenever $\mathcal{E}^{i,m}(t, \alpha) > \delta_2$, the grid adaption is carried out only once and at the same time $\mathcal{E}^{i,m}(t, \alpha)$ is reset to δ_1 for the determination of α on the refined mesh. Here, δ_1 and δ_2 are the prescribed lower and upper tolerance parameters for bounding the *a posteriori* error.

The above aim has the following detailed description for each time interval $(t_1, t_{1+1}]$ of total time interval $[t_0, t_f]$.

The adaptive algorithm seeks to construct a mesh $\Delta^{i,M}$ with mesh size vector $\bar{h}^{i,M}$ and the corresponding FE discrete solution

$$\bar{U}_1^{i,M}(x,t); \quad (1 \leq i \leq NPDE)$$

such that

$$\delta_1 \leq \varepsilon^{i,M}(t,\alpha) \leq \delta_2 \quad \text{or} \quad \mathcal{R}_{REL}^{i,M}(t) \leq \delta_{REL} \quad \text{for } t \in (t_i, t_{i+1}] \quad (5.12a)$$

where α is known apriorily or determined from the relation for all adaptions $m = 1, \dots, M$

$$\varepsilon^{i,m}(t,\alpha) - \delta_1 = 0 \quad (5.12b)$$

This is achieved by constructing a sequence of meshes

$$\Delta^{i,1}, \Delta^{i,2}, \Delta^{i,3}, \dots, \Delta^{i,M} \quad (5.12c)$$

of mesh size vector

$$\bar{h}^{i,1}, \bar{h}^{i,2}, \bar{h}^{i,3}, \dots, \bar{h}^{i,M} \quad (5.12d)$$

with corresponding computed solutions

$$\bar{U}_1^{i,1}(x,t), \bar{U}_1^{i,2}(x,t), \dots, \bar{U}_1^{i,M}(x,t); \quad \bar{U}^{i,m}(x,t) \in \mathcal{S}^{i,m}(\Delta) \quad (5.12e)$$

where $\Delta^{i,m+1}$ is constructed from

$$\bar{U}^{i,m}(x,t) \in \mathcal{S}^{i,m}(\Delta) \quad (5.12f)$$

by equi-distribution of the elemental error indicator contributions

$$E_n^{i,m}(t,\alpha); \quad (1 \leq n \leq N_e^{i,m})$$

to the *a posteriori* error

$$\varepsilon^{i,m}(t,\alpha), \quad t \in (t_i, t_{i+1}].$$

Here, the solution $\bar{U}^{i,m}(x,t)$ is obtained by solving the

semidiscrete problem

$$\begin{aligned}
 & \left\langle \frac{\partial}{\partial t} U_1^{i,m}(x,t), x_1^{i,m} \right\rangle + \left\langle a_1 \frac{\partial}{\partial x} U_1^{i,m}(x,t), \frac{\partial}{\partial x} x_1^{i,m} \right\rangle \\
 & + \nu \left\langle \frac{\partial}{\partial x} U_1^{i,m}(x,t), x_1^{i,m} \right\rangle = \left\langle \tilde{f}_1(x,t, U_1^{i,m}(x,t)), x_1^{i,m}(x,t) \right\rangle \\
 & \forall U_1^{i,m}(x,t) \in \mathcal{S}^{i,m}(t) \quad \forall x_1^{i,m}(x,t) \in \overline{\mathcal{F}}^{i,m}(t) \\
 & (1 \leq i \leq \text{NPDE}), \text{ and } m = 1, \dots, M \\
 & (x,t) \in \Omega \times (t_i, t_{i+1}] \\
 & \hspace{25em} (5.12g)
 \end{aligned}$$

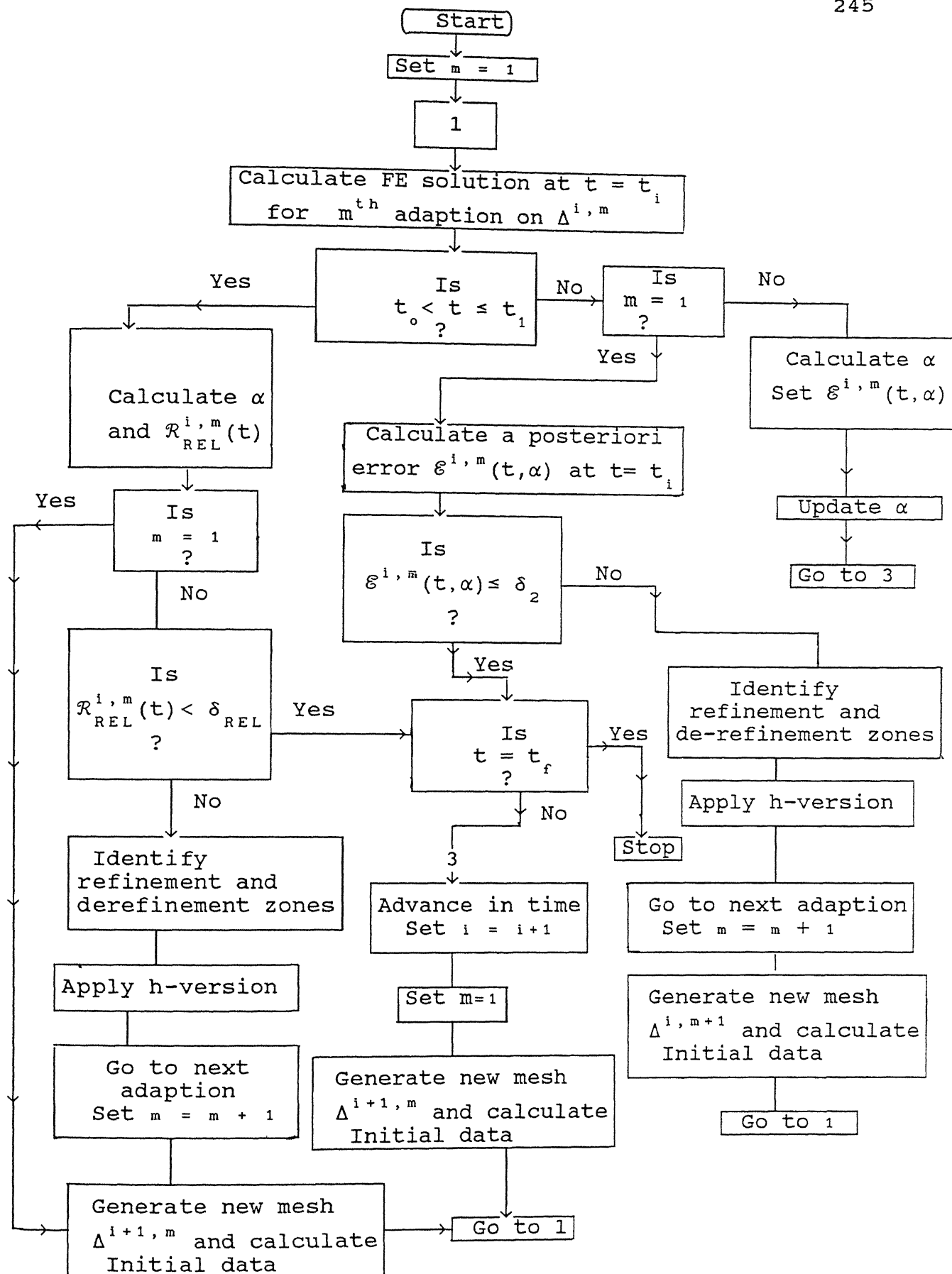
with appropriate boundary and initial conditions. In the equation (5.12a), the stopping criterion is based on residue for initial times $(t_0 < t \leq t_1]$ and on *a posteriori* error for later times $(t_1 < t \leq t_f]$.

5.6. h-version α strategy :

In this section, the α strategy based on h-version grid refinement is presented. Various steps involved in the finite element solution and the mesh modification are described in detail.

5.6.1. Description of the algorithm :

The timewise evolution of the FE solution and the mesh distribution with time or adaption level are described here. At the beginning of each time step, the initial data has been specified as described in the h version algorithm of Chapter 4. After each adaption, the mesh distribution and the starting guess solution change, while the initial condition requires interpolation to the new mesh. On the basis of the computed

Fig. 5.2. Flow chart for h-version based α strategy

finite element solution, a *a posteriori* error (or the relative global residue for the initial time step) is estimated with the help of local elemental error indicators . The decisions for mesh modification are then generated by the equi-distribution of the local error indicator $E_n(t, \alpha)$. For the initial time interval the adaptive grid is applied till two successive adaptions satisfy the tolerance criterion of eq. 5.10a. Here for each adaption the value of α is calculated and the smallest value of α is used in the error indicator expression for the mesh refinement decisions.

At later times, whenever the *a posteriori* error exceeds the upper tolerance parameter (δ_2) grid adaption begins, using the currently available value of the power index α for evaluating the error indicators through equation (5.8b). The grid adaption is done only once. On the refined mesh the finite element solution is calculated and the new value of α is evaluated from the eq. (5.11b). Further, at this stage, the value of α is updated by choosing the the smallest value from the previously available and the newly calculated values. This updated α is used while calculating the *a posteriori* error $\mathcal{E}(t, \alpha)$ at $t = t_{i+1}$. Also, the *a posteriori* error is reset equal to to lower tolerance parameter (δ_1). The solution at this stage is accepted as the computed solution for the updated time level. At later times, grid adaption is not carried out whenever the *a posteriori* error lies between two tolerance limits. The sequence of steps performed for each adaption cycle are enumerated below. These steps are also shown in the Fig. 5.2, in the form of flow chart.

Step 1 : Calculate the finite element solution at $t = t_i$

Calculate the FE solution at $t = t_i$ for m^{th} adaption on the mesh $\Delta^{i,m}$ with given initial condition and starting guess solution.

Step 2 : Check for nature of adaption for given time interval

If $t \in (t_0, t_1]$ then go to step 3a for the evaluation of relative global residue $\mathcal{R}_{\text{REL}}^{i,m}(t)$ and the index α .

If $t \notin (t_0, t_1]$ then go to step 3b for the evaluation of a posteriori error $\mathcal{E}^{i,m}(t, \alpha)$ and determination of α .

Step 3a : Calculate $\mathcal{R}_{\text{REL}}^{i,m}(t)$, index α and $E_n^{i,m}(t, \alpha)$, ($1 \leq n \leq N_e^{i,m}$)

Calculate $\mathcal{R}_{\text{REL}}^{i,m}(t)$ from equation (5.9b), compute index α from equation (5.10b) and elemental error indicators $E_n^{i,m}(t, \alpha)$ and go to step 4a. (For $m = 1$, choose $\mathcal{R}_{\text{REL}}^{i,m-1}(t) = 0$)

Step 3b : Calculate $\mathcal{E}^{i,m}(t)$, index α and $E_n^{i,m}(t, \alpha)$, ($1 \leq n \leq N_e^{i,m}$)

If $m = 1$ (starting adaption) then

Calculate the elemental errors $E_n^{i,m}(t, \alpha)$, ($1 \leq n \leq N_e^{i,m}$)

Calculate the a posteriori error estimate $\mathcal{E}^{i,m}(t, \alpha)$

with updated α and go to step 4b.

If $m = 2$ (final adaption) then

solve for new α from the eq. $\mathcal{E}^{i,m}(t, \alpha) - \delta_1 = 0$,

reset the a posteriori error $\mathcal{E}^{i,m}(t, \alpha)$ to δ_1 .

Select the least of new and previous values of α

and go to step 7 for time marching.

Step 4 : Check for the application of adaptive grid .

(a) : if $t \in (t_0, t_1]$ and $\mathcal{R}_{\text{REL}}^{i,m}(t) \leq \delta_{\text{REL}}$ then stop the adaption and go to step 7 for time marching.

if $t \in (t_0, t_1]$ and $\mathcal{R}_{\text{REL}}^{i,m}(t) > \delta_{\text{REL}}$ then go to step 5

for adaption.

(b): if $t \in (t_1, t_f]$ and $\delta_1 \leq \varepsilon^{i,m}(t, \alpha) \leq \delta_2$ then

go to step 7 for time marching.

if $t \in (t_1, t_f]$ and $\varepsilon^{i,m}(t, \alpha) > \delta_2$ then

go to step 5 for adaption.

Step 5.: Implementation of adaptive h-refinement scheme

(i). For for any $\Omega_n^{i,m} \in \Delta^{i,m}$, if $\text{MINTOL} < E_n^{i,m}(t, \alpha) < \bar{\nu}^{i,m}$ then **RETAIN** the element as it is. Here, $\bar{\nu}^{i,m}$ is the mean value of the error indicators $E_n^{i,m}(t, \alpha)$ and **MINTOL** is the specified tolerance for node removal. **MINTOL** is dynamically set as

$$\text{MINTOL} = d * \bar{\nu}^{i,m} \text{ where } (0 < d < 1).$$

(ii). For any pair of elements, $(\Omega_n^{i,m}, \Omega_{n+1}^{i,m})$ if $E_n^{i,m}(t, \alpha) \wedge E_{n+1}^{i,m}(t, \alpha) \leq \text{MINTOL} < \bar{\nu}^{i,m}$, then **REMOVE** the common node between these elements.

(iii). For for any $\Omega_n^{i,m} \in \Delta^{i,m}$, if $E_n^{i,m}(t, \alpha) > \bar{\nu}^{i,m}$ then **REFINE** the element according to statistical procedure.

(iv). Perform the above steps 5 (i), (ii), (iii) and (iv) for all elements on the mesh .

(v). Generate initial data on the new mesh $\Delta^{i, m+1}$ increase the adaption counter (m) by one, and go to step 1.

Step 6 :Time marching :

if $t < t_f$ then set $t_{i+1} = t_i + \Delta t$; construct and generate the initial data on the coarse mesh $\Delta^{i+1, 1}$ at $t = t_{i+1}$, reset counter (m) to 1 and go to step 1
if $t = t_f$ then stop the algorithm.

5.6.2. Important features of Algorithm :

The procedure for evaluating the initial condition, starting guess solution at any adaption and time level are similar to those used in the h -refinement algorithm of Chapter 4. Hence the details of these procedures are not discussed here.

Finite element solution :

The matrix contributions $[\tilde{M}]$, $[\tilde{K}^1]$ and $[\tilde{K}^2]$ in the equation (5.7b) are due to the additional terms in the test functions with quadratic bias. These modifications arising from upwinding formulation are easily implemented at the element level and the resulting matrix system (5.7b) is solved by the iterative method described in Chapter 3.

Decisions for mesh refinement :

In the following section, the adaptive mesh modification based on the newly proposed *a posteriori* error estimate and some other important aspects of the adaptive mesh generation procedure are discussed.

For the initial time interval, the adaptive grid refinement is carried out based on the equi-distribution of local error indicators, till the global residue for two successive adaptions satisfies the eq. 5.10a. After a sufficient number of adaptions the global residue and the index α attain stable values.

For subsequent time intervals, when the *a posteriori* error exceeds the upper tolerance δ_2 , mesh refinement is performed according to statistical procedure and the value of α is updated

on the refined mesh using equation (5.11b). Also, the *a posteriori* error is set equal to the lower tolerance limit δ_1 . Grid adaption is not carried out whenever the *a posteriori* error lies between two bounds. In actual computations, at later times, the number of adaption steps is always restricted to two, since the *a posteriori* error turns out to be nearly equal to the lower tolerance parameter after one adaption. The evolution of a *a posteriori* error with adaption and time level is shown schematically in Fig. 5.3.

In the de-refinement process, the criterion for the removal of nodes is set dynamically as a given fraction of the mean of local error indicators. A constant 'd' ($0 < d < 1$) has been chosen such that the criterion for the removal of nodes is given by $\text{MINTOL} = d * \bar{\epsilon}^{1,m}$. A low value of the constant will result in slow removal of nodes in comparison to the stuffing of nodes. In the computations of the present study the value of d has been chosen as 0.75.

Calculation of index α :

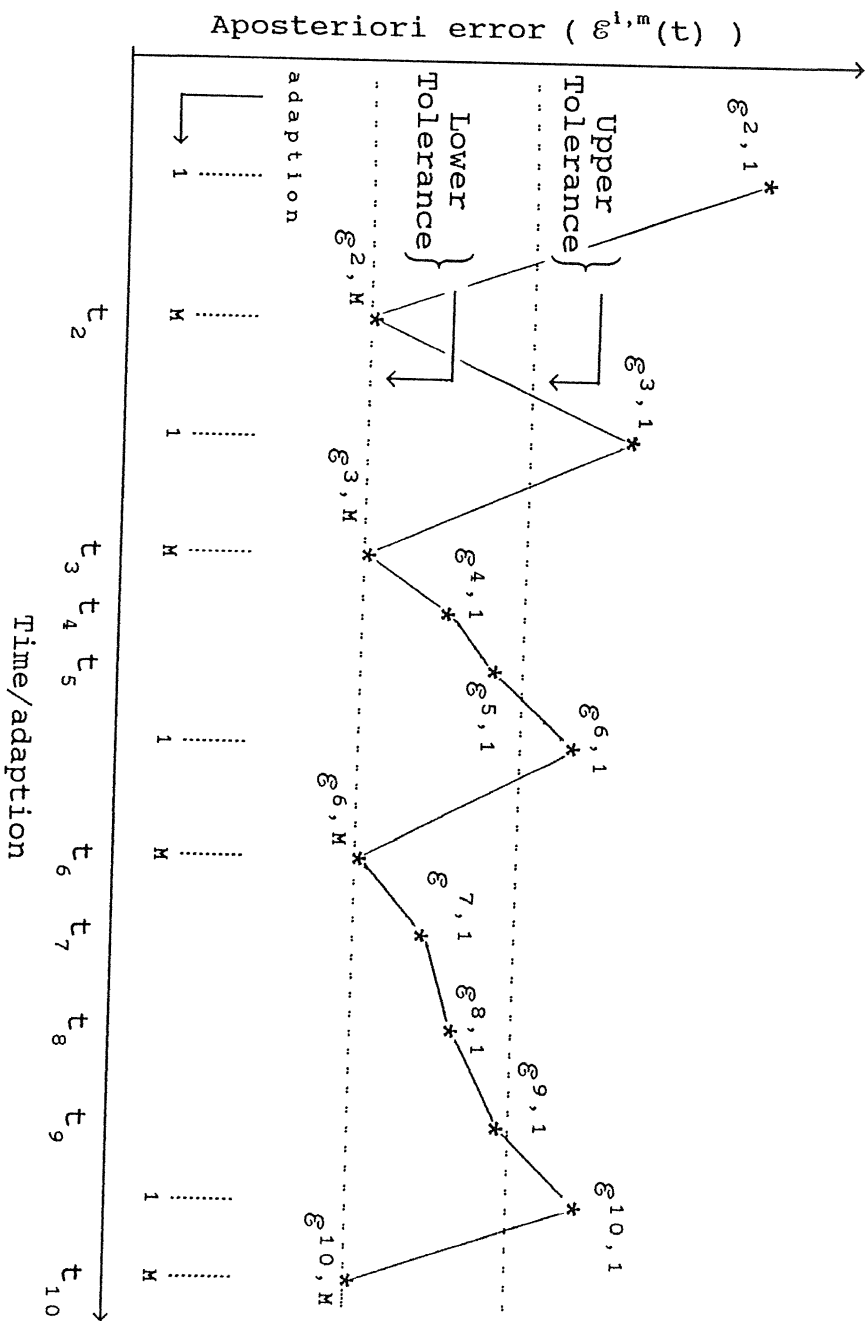
Let us suppose that the adaptive grid has been applied on the mesh $\Delta^{1,m}$ at $t = t_1$ for the m^{th} adaption. We calculate the index α from the equation

$$\epsilon^{1,m}(t, \alpha) - \delta_1 = 0$$

$$\text{i.e., } \sum_{n=1}^{N_e^{1,m}} \sum_{l=1}^{NPDE} h_n^\alpha \int_{\Omega_n^{1,m}} r_{1,n}^{1,m}(s, t)^2 ds - \delta_1 = 0 = y(\alpha) \quad (\text{say}) \quad (5.13)$$

The root α is calculated by the bisection method. Since the step

Fig. 5.3. Schematic representation of the evolution of a posteriori error with adaptions/time in h version based α strategy.



Results have been predicted for various values of flow velocity. The situation without flow ($V = 0$) has also been considered as a special case, for the sake of comparison with the results presented in Chapter 4. The nature of the solution profiles, behaviour of the global and local error measures and the variation of the exponent α are discussed.

5.7.1. Application of α strategy to the test problem :

(a). Solution profiles and grid distributions :

The exact and the computed solutions of the test problem are compared at various time levels in Fig. 5.4a-c. for three different flow velocities. The same exact solutions have been used for all these velocities, by selecting the source term $\bar{f}(x,t)$ of eqn. (5.1) appropriately. It is observed that the exact and the computed solutions match excellently at each time, during the initial transient stage as well as the uniform movement phase of the front. For the first few time steps, in order to compensate for the inaccuracies arising from the coarse mesh, the initial condition for each time step has been taken from the corresponding exact solutions.

The front shape and its movement are predicted very well by the α -strategy analogous to the refinement strategies based on the theoretical error estimates of Bieterman and Babuska (1982a,b, 1986) described in Chapter 4. However, it should be noted that these theoretical error estimates are not applicable to the non-linear non self adjoint operator, of the present problem. In this sense, the α -strategy could be a powerful tool for generating meshes in an adaptive manner for any general problem. The tables

associated with Fig. 5.4a-c show the number of elements at various time levels. At time $t = 0$, a uniform mesh of 10 elements has been used in each case. It is seen that the number of elements increases rapidly during the initial transient stage and rather slowly when the front has a uniform movement phase. These features of grid refinement are closely related to the improved estimation of the exponent α with each grid adaption. Indeed the α -estimate improves rapidly during the initial adaptations and slowly in the later stages. More discussion on these trends of the exponent is presented subsequently. During the initial time interval, the computed solution is accepted if the relative residue satisfies the tolerance (10^{-2}) criterion.

In general, the number of elements at any time is somewhat less for the α -strategy as compared to the procedures discussed in Chapter 4. With respect to the flow velocity variation, the number of elements is more or less the same for low velocity values. For high flow velocity values ($V = 10.0$), however, the number of elements is significantly more. The grid distributions for various flow velocity values and time instants are shown in Fig. 5.5a-c. It is evident from the figure that the α -strategy provides dense refinement in the front zone and that the grid distribution faithfully follows the front movement with respect to time. It appears, however, that the magnitude of error is different for each flow velocity; this may be the reason why the number of elements is not the same, although the same exact solution has been used in all the cases. The plots also indicate that grid refinement occurs in anticipation of front arrival in regions ahead of the front, while gradual derefinement occurs in

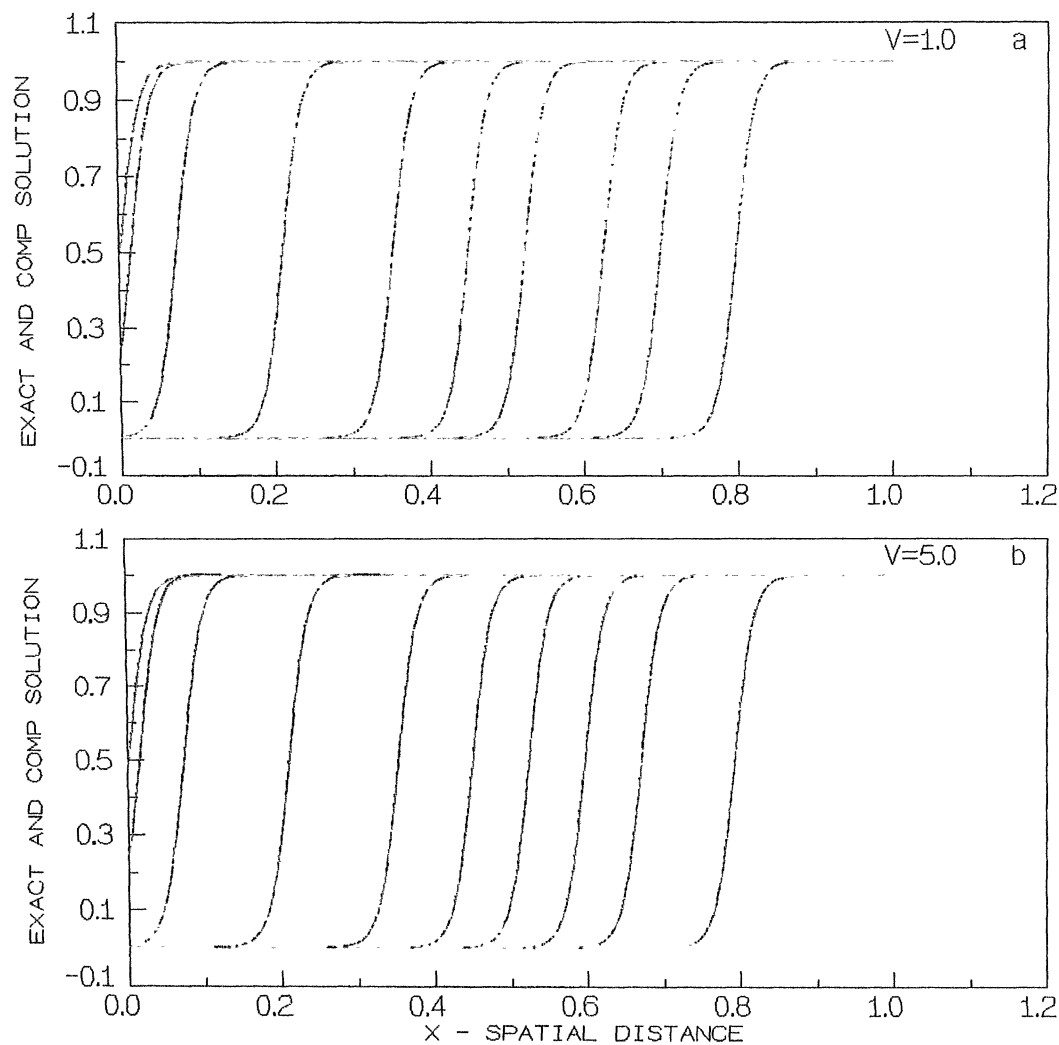


Fig. 5.4 TEST PROBLEM (h version) : SOLUTION PROFILE

| Solution | Line_type |
|------------|-----------|
| Comp soln | ----- |
| Exact soln | ----- |

Fig. (a)

| Adp | Time | Eles | Nods |
|-----|-----------|------|------|
| 1 | 0 .000050 | 21 | 22 |
| 2 | 0 .001400 | 36 | 37 |
| 3 | 0 .007250 | 92 | 93 |
| 4 | 0 .021000 | 133 | 134 |
| 5 | 0 .035250 | 161 | 162 |
| 6 | 0 .044950 | 161 | 162 |
| 7 | 0 .052300 | 187 | 188 |
| 8 | 0 .062500 | 221 | 222 |
| 9 | 0 .069850 | 205 | 206 |
| 10 | 0 .079650 | 229 | 230 |

Fig. (b)

| Adp | Time | Eles | Nods |
|-----|-----------|------|------|
| 1 | 0 .000050 | 20 | 21 |
| 2 | 0 .001400 | 25 | 26 |
| 3 | 0 .007250 | 58 | 59 |
| 4 | 0 .021000 | 90 | 91 |
| 5 | 0 .035250 | 112 | 113 |
| 6 | 0 .044950 | 156 | 157 |
| 7 | 0 .052300 | 159 | 160 |
| 8 | 0 .059650 | 170 | 171 |
| 9 | 0 .067000 | 174 | 175 |
| 10 | 0 .079250 | 214 | 215 |

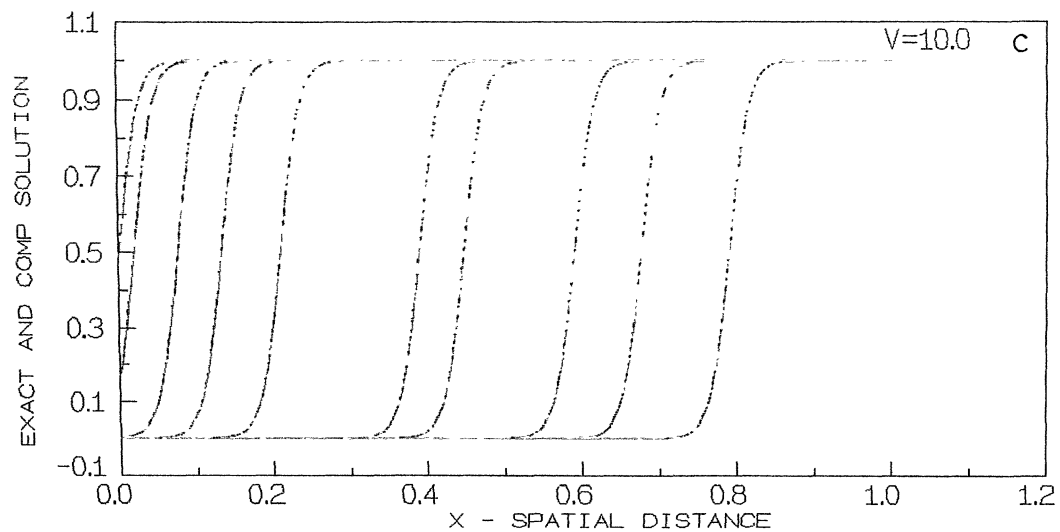


Fig. 5.4 TEST PROBLEM (h version) : SOLUTION PROFILE

| Solution | Line_type |
|------------|-----------|
| Comp soln | ----- |
| Exact soln | - - - - - |

Fig. (a)

| Adp | Time | Eles | Nods |
|-----|-----------|------|------|
| 1 | 0 .000050 | 27 | 28 |
| 2 | 0 .001950 | 34 | 35 |
| 3 | 0 .007650 | 91 | 92 |
| 4 | 0 .013350 | 122 | 123 |
| 5 | 0 .021000 | 121 | 122 |
| 6 | 0 .039050 | 227 | 228 |
| 7 | 0 .044800 | 245 | 246 |
| 8 | 0 .059050 | 270 | 271 |
| 9 | 0 .067650 | 327 | 328 |
| 10 | 0 .079050 | 338 | 339 |

Fig. (b)

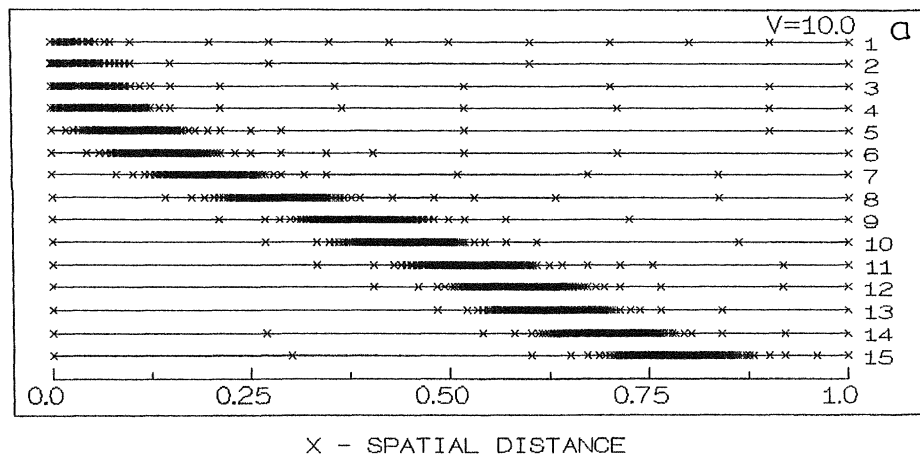


Fig. 5.5 TEST PROBLEM (h version) : ADAPTED MESH AT VARIOUS t

| Adp | Time | Eles | Nods |
|-----|-----------|------|------|
| 1 | 0 .000050 | 27 | 28 |
| 2 | 0 .001950 | 34 | 35 |
| 3 | 0 .003850 | 54 | 55 |
| 4 | 0 .006700 | 79 | 80 |
| 5 | 0 .010500 | 111 | 112 |
| 6 | 0 .015250 | 122 | 123 |
| 7 | 0 .020050 | 121 | 122 |
| 8 | 0 .029550 | 172 | 173 |
| 9 | 0 .039050 | 227 | 228 |
| 10 | 0 .044800 | 245 | 246 |
| 11 | 0 .053350 | 264 | 265 |
| 12 | 0 .059050 | 270 | 271 |
| 13 | 0 .063850 | 315 | 316 |
| 14 | 0 .070500 | 323 | 324 |
| 15 | 0 .079050 | 338 | 339 |

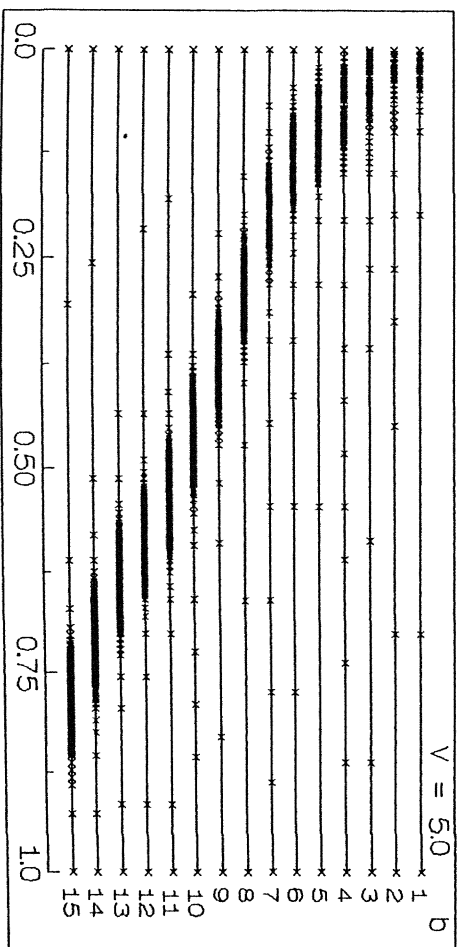
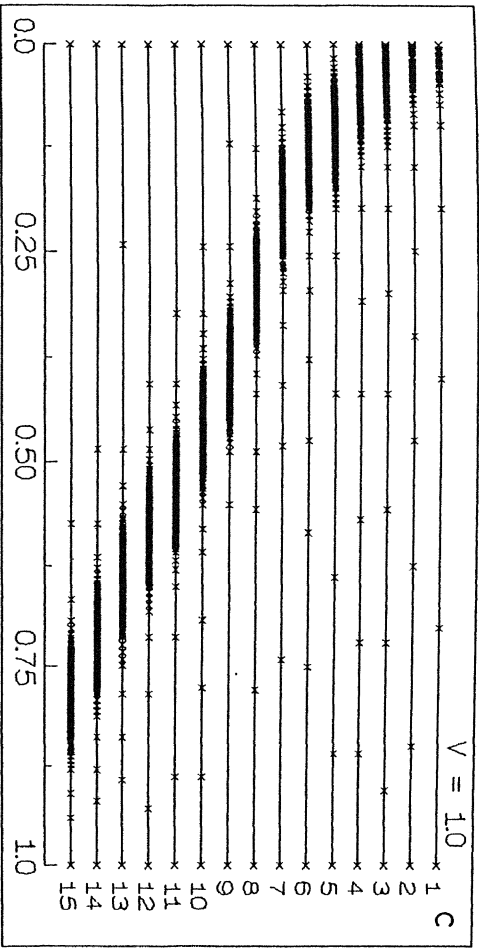


Fig. 5.5 TEST PROBLEM (h version) ADAPTED MESH AT VARIOUS t

| Adp | Time | Eles | Nods |
|-----|-----------|------|------|
| 1 | 0 .000050 | 20 | 21 |
| 2 | 0 .001400 | 25 | 26 |
| 3 | 0 .003650 | 42 | 43 |
| 4 | 0 .006800 | 58 | 59 |
| 5 | 0 .010400 | 66 | 67 |
| 6 | 0 .014450 | 77 | 78 |
| 7 | 0 .021000 | 90 | 91 |
| 8 | 0 .029550 | 97 | 98 |
| 9 | 0 .039050 | 126 | 127 |
| 10 | 0 .047400 | 158 | 159 |
| 11 | 0 .054750 | 161 | 162 |
| 12 | 0 .059650 | 170 | 171 |
| 13 | 0 .064550 | 179 | 180 |
| 14 | 0 .071900 | 186 | 187 |
| 15 | 0 .079250 | 214 | 215 |



| Adp | Time | Elas | Nods |
|-----|-----------|------|------|
| 1 | 0 .000050 | 21 | 22 |
| 2 | 0 .001400 | 36 | 37 |
| 3 | 0 .003650 | 68 | 69 |
| 4 | 0 .006800 | 92 | 93 |
| 5 | 0 010400 | 126 | 127 |
| 6 | 0 .014450 | 136 | 137 |
| 7 | 0 .021000 | 133 | 134 |
| 8 | 0 .029550 | 149 | 150 |
| 9 | 0 .039050 | 149 | 150 |
| 10 | 0 .047400 | 167 | 168 |
| 11 | 0 .054750 | 190 | 191 |
| 12 | 0 .059650 | 204 | 205 |
| 13 | 0 .064950 | 225 | 226 |
| 14 | 0 .072300 | 217 | 218 |
| 15 | 0 .079650 | 229 | 230 |

X - SPATIAL DISTANCE

the regions behind the front. These features are highly essential for any adaptive strategy which is to be used for the capturing of mobile high gradient fronts.

The adaptive algorithm automatically removes, retains and stuffs nodes simultaneously. It is however, possible that some unnecessary nodes exist in regions where no solution activity is taking place, as observed from the Figs. 5.6a-b. Also, at few time instants, refinement is slightly less dense, even in the flame front region which can be seen from the figures. Except for some minor exceptions, the figures clearly illustrate the ability of the adaptive scheme to concentrate grid points around the characteristic wave, where all the activity takes place. The side tables provide the information of refinement and de-refinement of elements in the adaption.

In the present work, removal of nodes has been carried out if the error of the neighbouring elements goes below a prescribed fraction of the mean error. Thus, the minimum tolerance limit for removal of nodes in the passive regions, is set dynamically. The removal of nodes in the de-refinement process is very fast if the fraction of mean error (d) is chosen appropriately. The value of fraction d has been taken as 0.75. Due to this relatively large cut off value, however, mesh modification is not done properly at some time instants, although the trend is alright and high gradient regions are refined as expected (Fig. 5.6a). On the other hand, over refinement of the grid occurs some times due to the dense stuffing of nodes by the statistical refinement procedure. However, this can easily be handled by specifying a minimum grid spacing. Further, for the first few adaptations, the resolution of

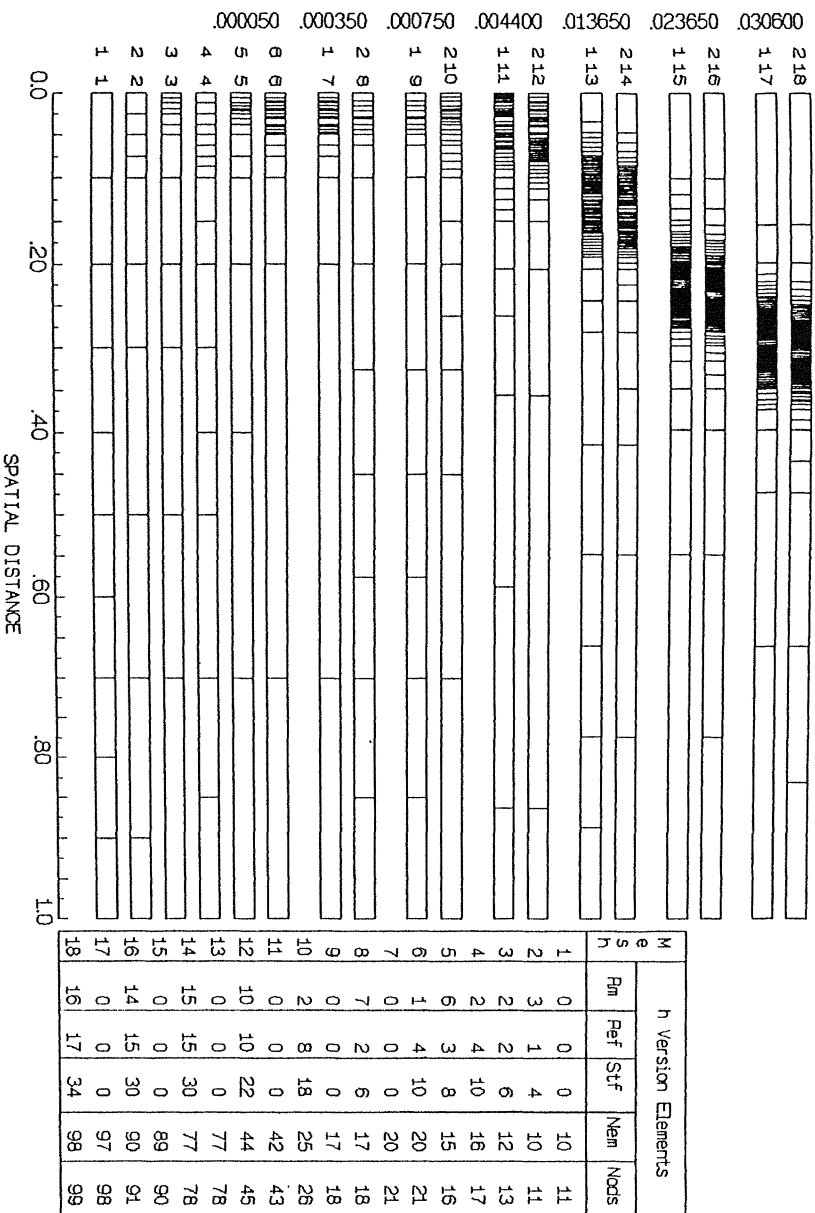


FIG.5.6(a) TEST PROBLEM : ADAPTED MESH AT VARIOUS TIMES (V=5.0)

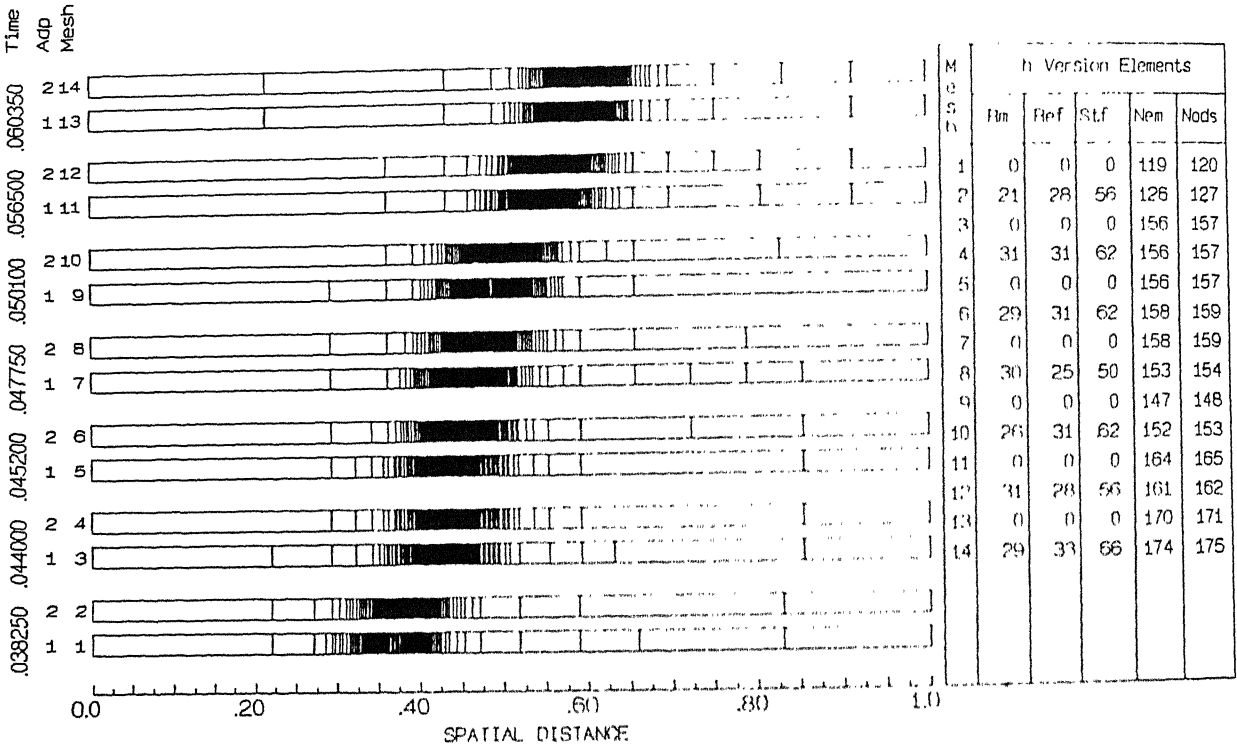


FIG.5.6(b) TEST PROBLEM : ADAPTED MESH AT VARIOUS TIMES (V = 5.0)

front region is not proper because of coarse initial grid. The main reason for this is the improper evaluation of the index α which in turn significantly influences the magnitude of the local error indicator. However, this problem is eliminated in the later adaptations as the grid becomes more refined. (see Fig. 5.6a-b). At later stages of grid adaption, the mean error is very small and it is prone to be affected by round off error. Thus, at few instants, elements which need not be refined are also divided further. In the computations for the α strategy, a large time step value ($\Delta t \approx 10^{-4}$) has been used. Because of this, the local elemental errors do not behave perfectly well, especially adjacent to the moving front. Large values of Δt , cut-off fraction for defining MINTOL etc. have been purposefully used here, to subject the α strategy to a more stringent test. If these values are lowered suitably, excellent results can be obtained as in Chapter 4, at extra computational cost. It can be seen that the number of elements required to satisfy a similar error tolerance criterion as in Chapter 4, is much less for the α strategy. Figs. 5.6a-b, it is clear that the nodes are distributed in anticipation of the front motion.

(b). Behaviour of index α :

In Fig. 5.7, the exponent of α has been plotted against time, for three different flow velocities. The timewise evolution of α including the variations over each adaption step of the time increments Δt , is also depicted. It is observed that α has a high value in the beginning when the global residue (error) is large and the mesh is coarse. For such coarse initial meshes, a good

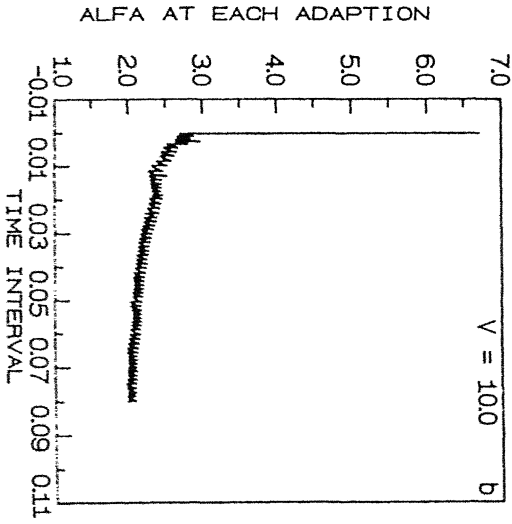
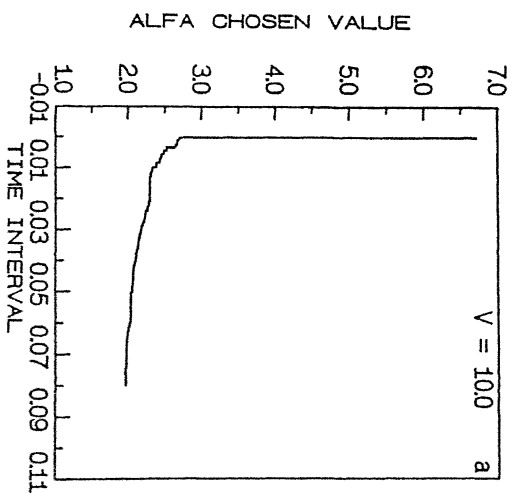
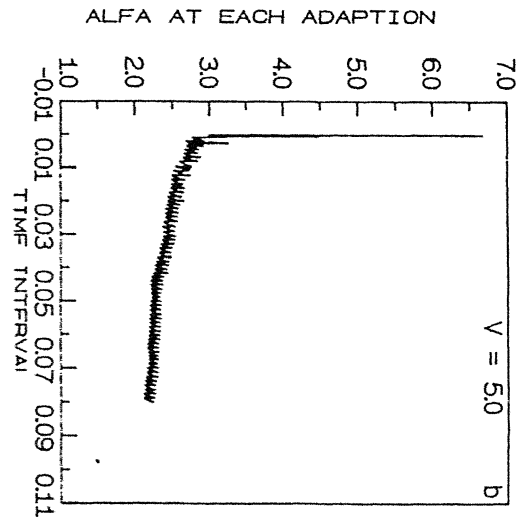
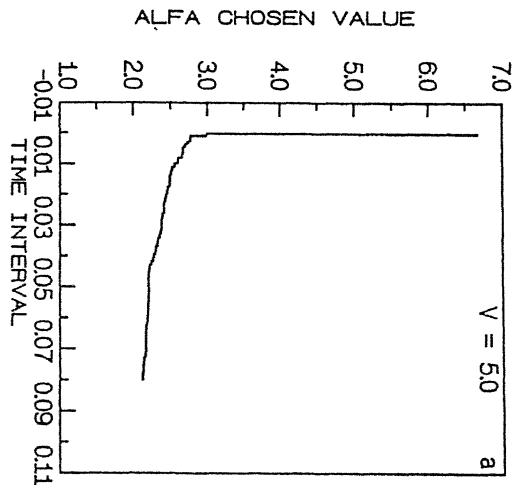
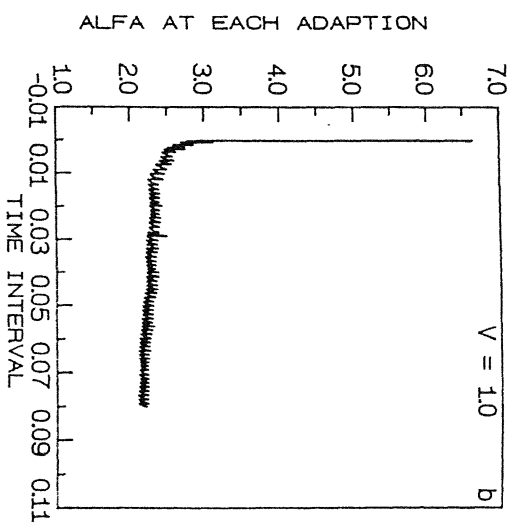
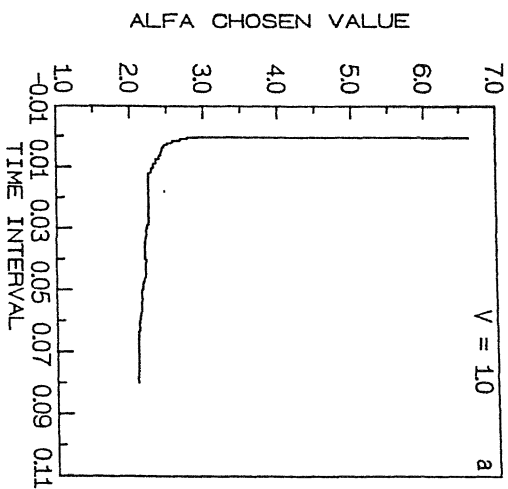


Fig. 5.7. TEST PROBLEM : TIME EVOLUTION OF ALFA VALUES (h VERSION)



value of α is obtained by using an adaption stopping criterion based on global relative residue. This is equivalent to reducing the error in the solution at initial time. With successive adaptions, the value of α decreases rapidly and attains a stable value for refined meshes. This process can be likened to the approach towards the asymptotically valid error estimates of Babuska and Rheinboldt (1982a,b) for $h \rightarrow 0$. Indeed for refined meshes, α approaches the value of 2, as in the error estimator expressions described in Chapter 4. This proves the self correcting feature of the α -strategy, which leads to improved estimation of the computable error after each grid adaption cycle, until the computable error approaches the true error for the problem. The minor oscillations seen in the α value with respect to adaption level within each time step, arise due to the front movement during the time increment Δt . Also, the choice of a large time ($\Delta t \approx 10^{-4}$) step may be another reason for these oscillations. In fact, the error tends to increase as the front moves out of an optimal grid distribution at a particular time instant. This increase in error tends to increase the α value slightly. However, in the next adaption cycle, the grid distribution is adjusted again to the new front location and the value of α is corrected back to more or less the original value. The overall variation of α with respect to time follows a monotonically decreasing trend, confirming the progressively improved error estimation feature of the present strategy. For all the flow velocities considered, the behavior of α essentially remains the same. These trends confirm the proposal of Szymczak and Babuska (1984) that in the asymptotic limit of $h \rightarrow 0$, the

computable error is well estimated by their error estimator expression for the linear convective-diffusive problems.

(c). Behaviour of residue and error estimates :

In Fig. 5.8, the variations of the *a posteriori* error and the gradient error with time are shown. Since two limits (δ_1 and δ_2) have been employed for starting and stopping the grid adaption, the *a posteriori* error oscillates with respect to time within these two bounds. Occasionally, during the course of a particular adaption or time step, the error marginally exceeds the upper limit or dips below the lower limit. The plots of gradient error indicate that for refined meshes, maintaining the *a posteriori* error within certain bounds implies the limiting of the gradient error as well. Moreover, as time proceeds, the mean value of the error decreases and eventually stabilizes.

In Fig. 5.9, the variations of the maximum and the mean local error indicators are plotted with time. The instantaneous values of these error indicators oscillate with respect to time, as a consequence of limiting the *a posteriori* error within two bounds. However, their average values decrease with time monotonically indicating successive improvement in the computed error estimate, the associated grid distribution and the predicted solution. The magnitudes of the maximum and the mean error indicators are almost of the same order, implying that the statistical procedure is very effective in equi-distributing the local error. With respect to flow velocity, the computed error values show some variation, although the range of minimum does not differ much.

The maximum and the mean values of the local elemental

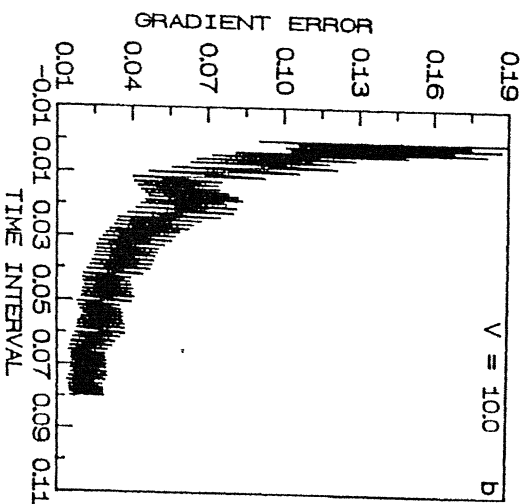
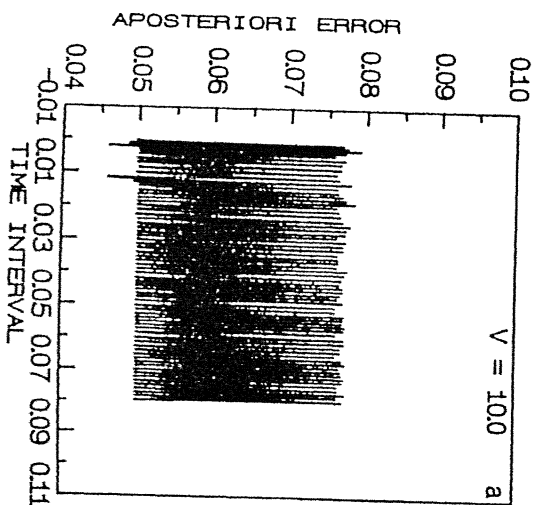
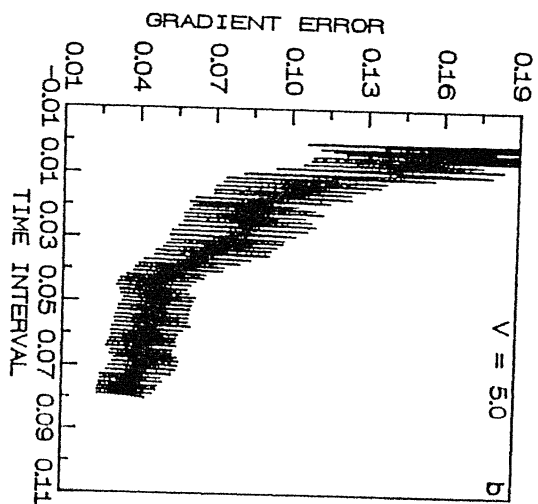
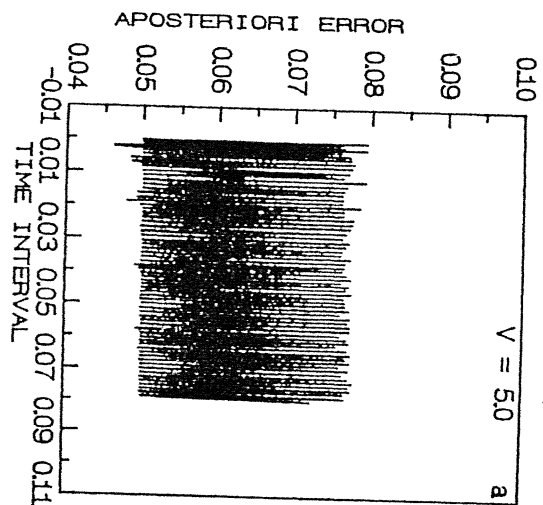
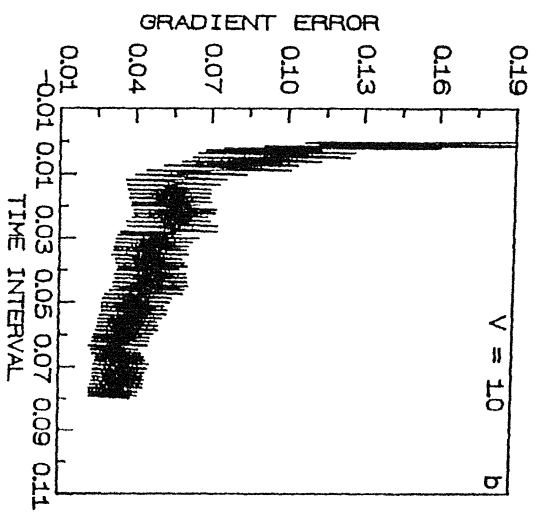
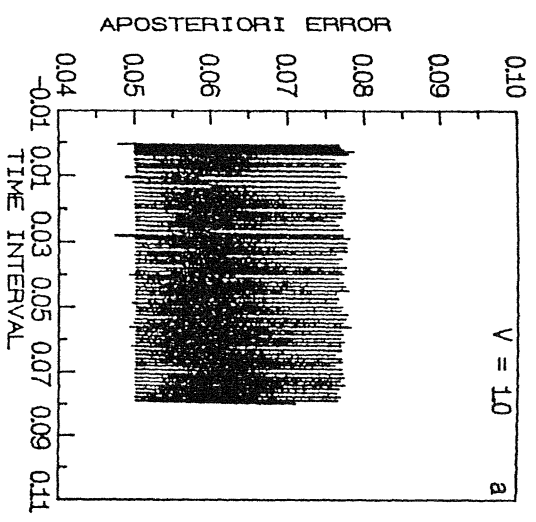


Fig. 5.8. TEST PROBLEM : TIME EVOLUTION OF ERRORS (h VERSION)
(APOSTERIORI ERROR AND GRADIENT ERROR)



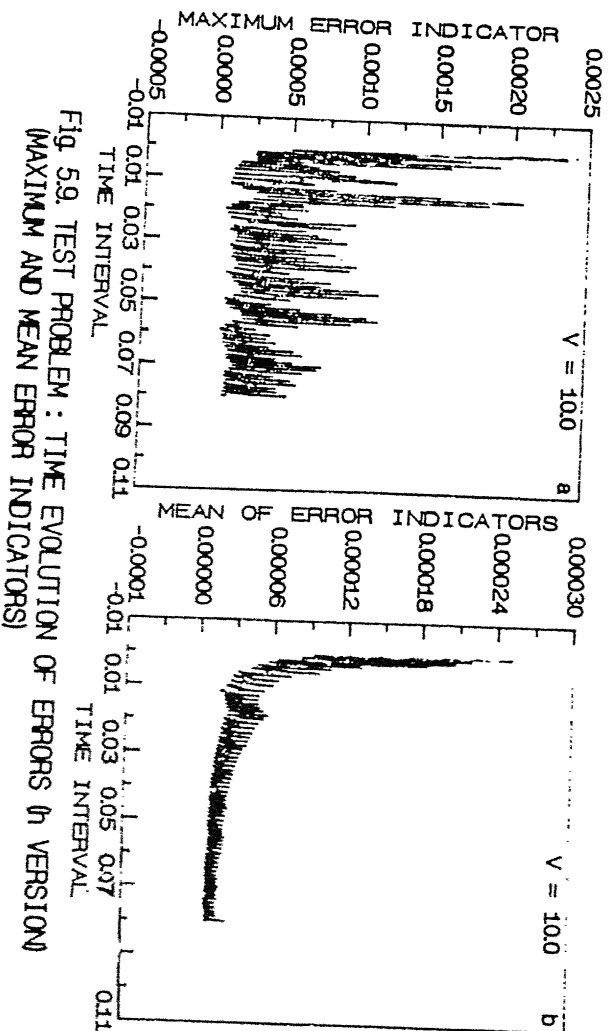
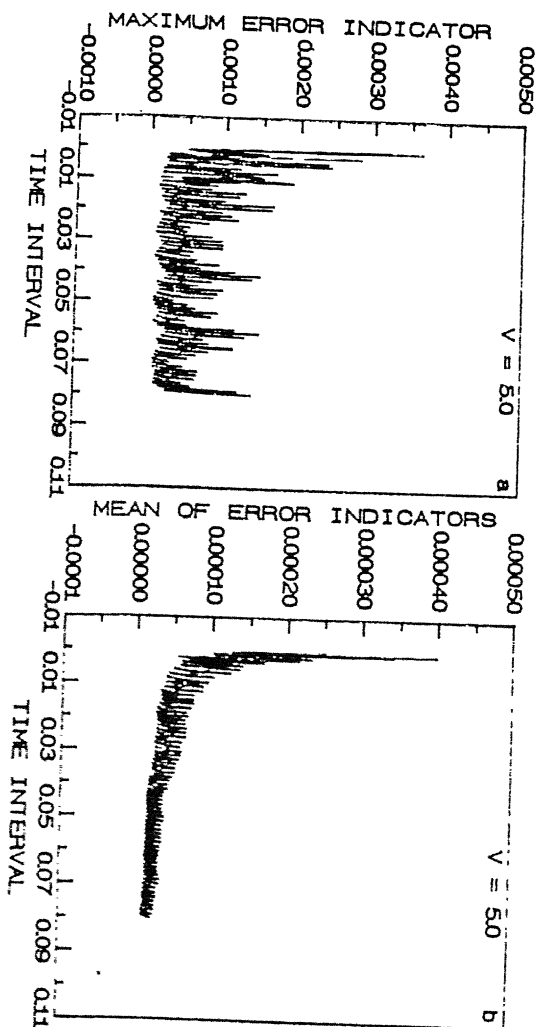
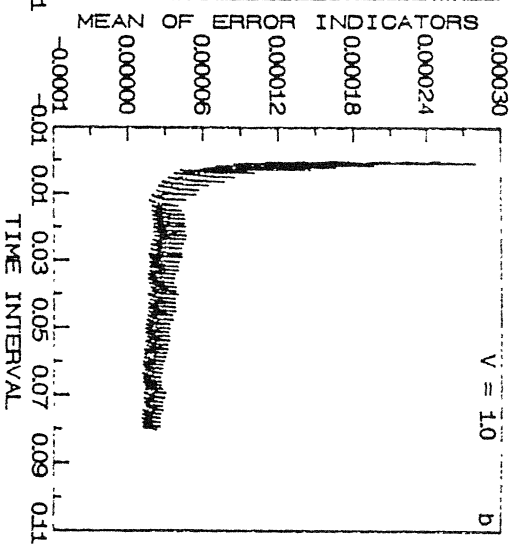
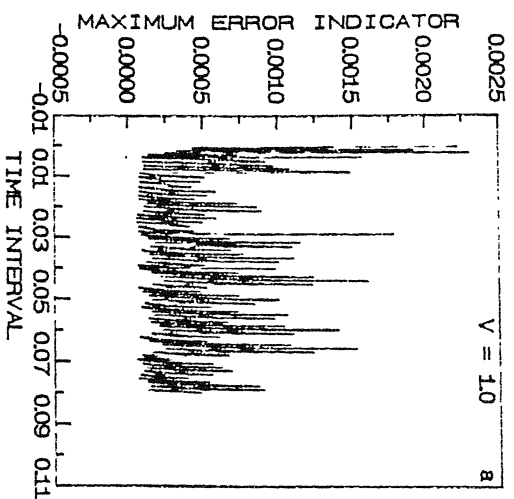


Fig 59. TEST PROBLEM : TIME EVOLUTION OF ERRORS (h VERSION)
(MAXIMUM AND MEAN ERROR INDICATORS)



residues are plotted against time in Fig. 5.10. These elemental residues also decrease rapidly in the beginning approaching towards stable values for refined meshes, similar to the features exhibited by the other error measures.

The variation of the global residue and the global solution gradient (in the respective norms) are shown in Fig. 5.11. For initial times, these norms increase due to the growth of the front. In the later stage, when the front is fully developed and its velocity is constant, the norms of the residue and the gradient also attain constant values.

The spatial distributions of the local error indicators are plotted in Figs. 5.12-5.14 for a flow velocity value of $V = 5.0$. These figures show that the magnitude of the local error indicator decreases with grid adaption, as expected. The number of adaptations required to meet the *a posteriori* error tolerance criterion is large in the beginning, due to the coarse initial mesh. For later times, however, just two adaptations are sufficient. The band of sharp error peaks (representing the zone which contributes primarily to the global error) moves with time, in response to a similar movement in the front location. A few error peaks of large width are also seen in the figures. at some time instants. These imply that one or two elements lying outside the front region have not been refined enough; but the contribution of such elements to the global error is negligible and it does not matter even if the local error has not been equi-distributed well over these elements.

(d). Behaviour of error measures for the case $V = 0$:

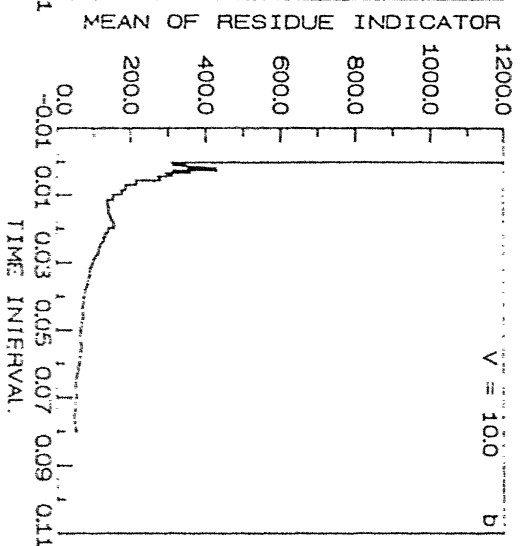
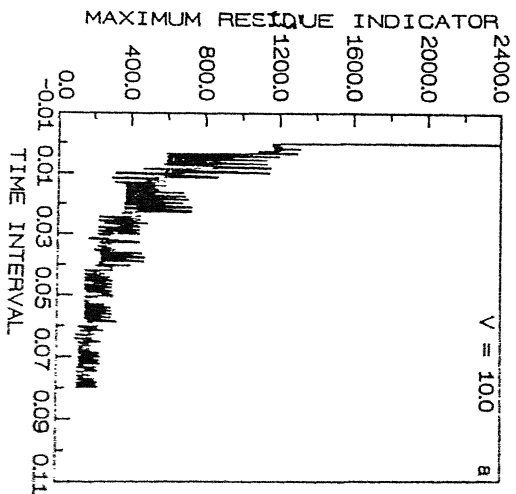
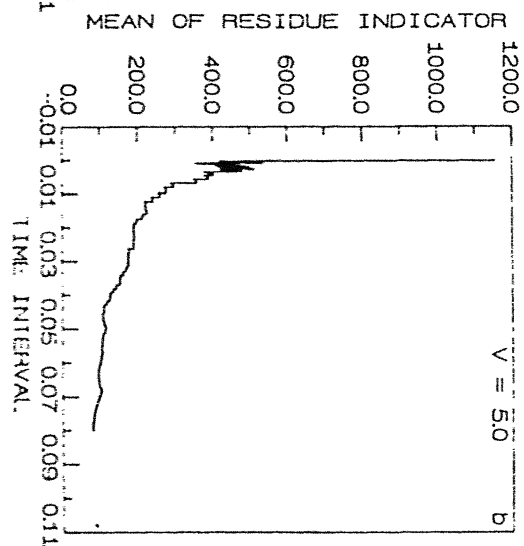
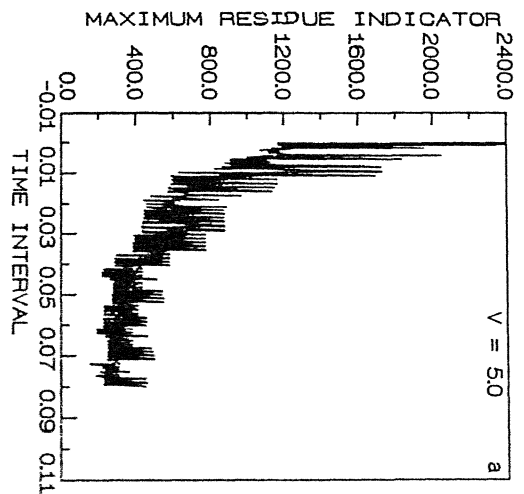
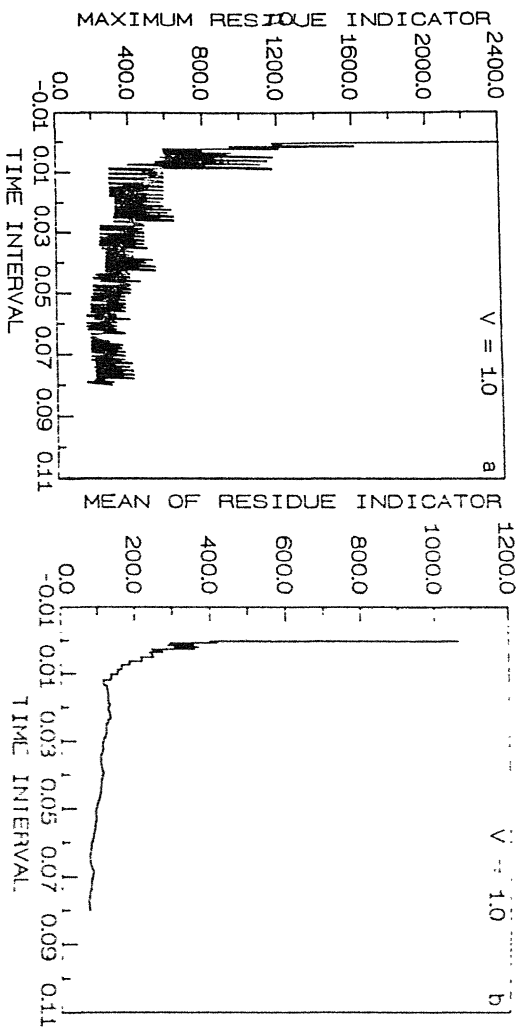


Fig. 5.10. TEST PROBLEM : TIME EVOLUTION OF RESIDUES (h VERSION)
(MAXIMUM AND MEAN RESIDUE INDICATORS)



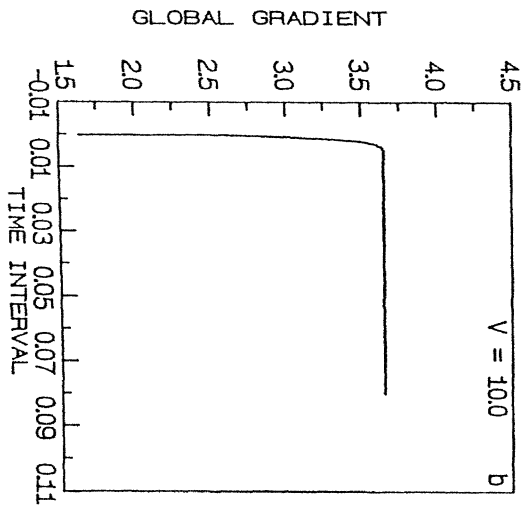
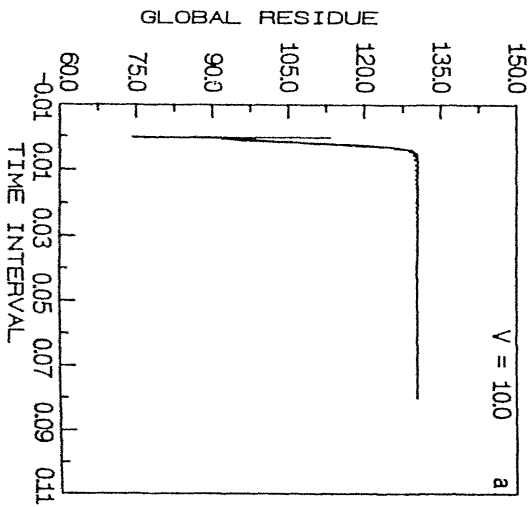
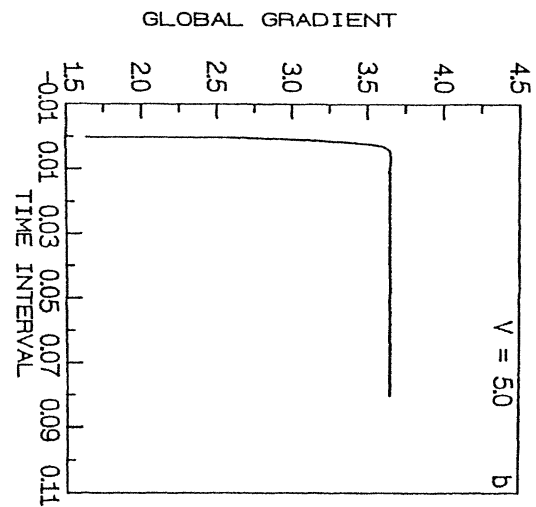
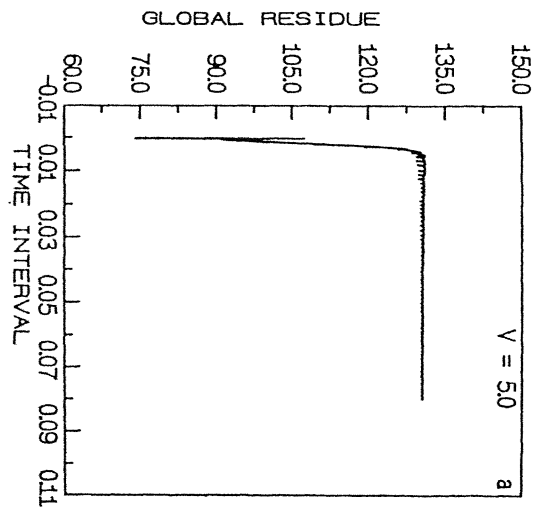
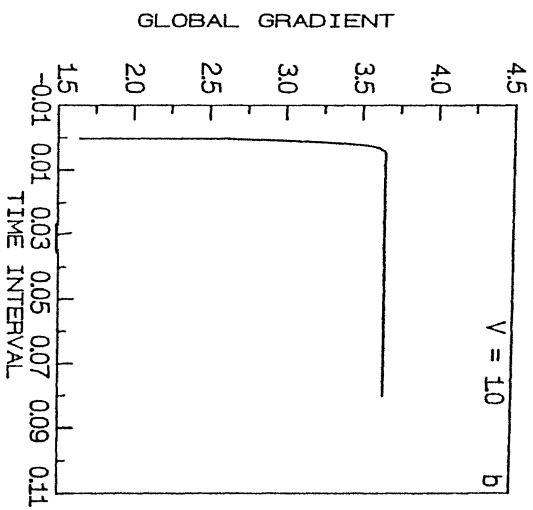
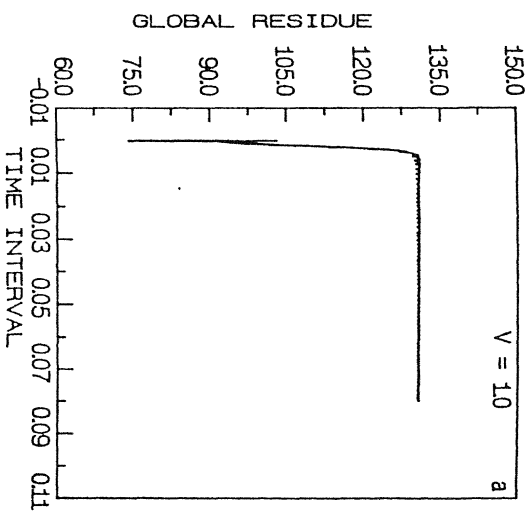
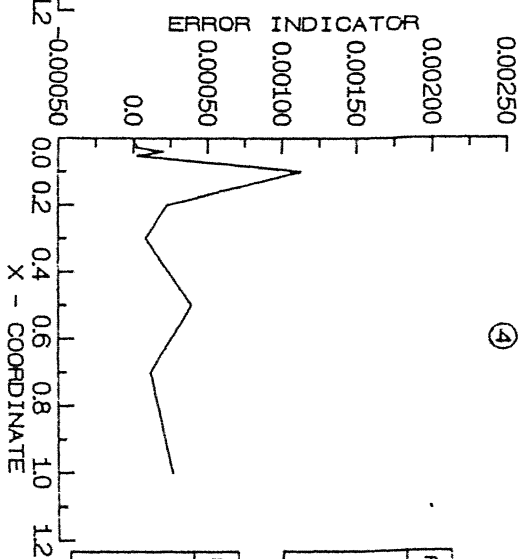
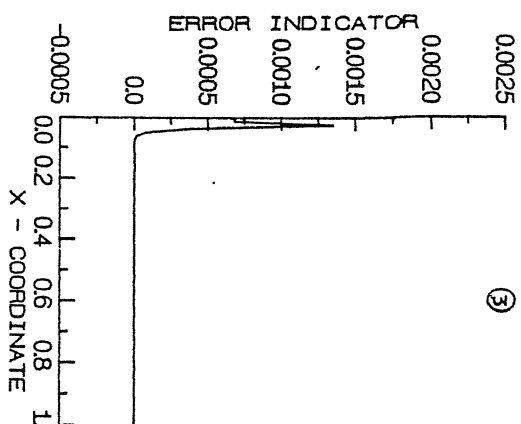


Fig. 5.11 TEST PROBLEM : TIME EVOLUTION OF GLOBAL RESIDUE AND GLOBAL GRADIENT (h VERSION)





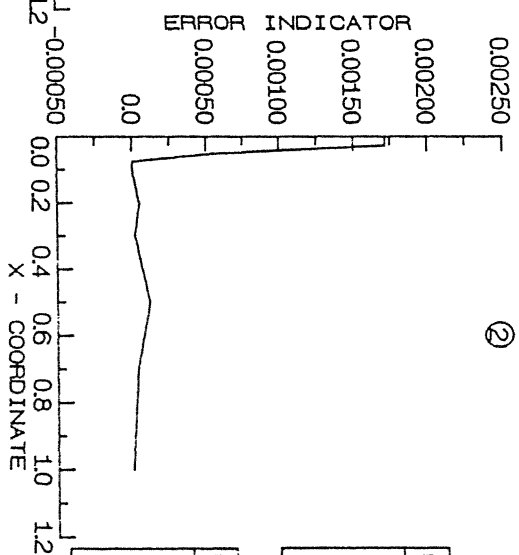
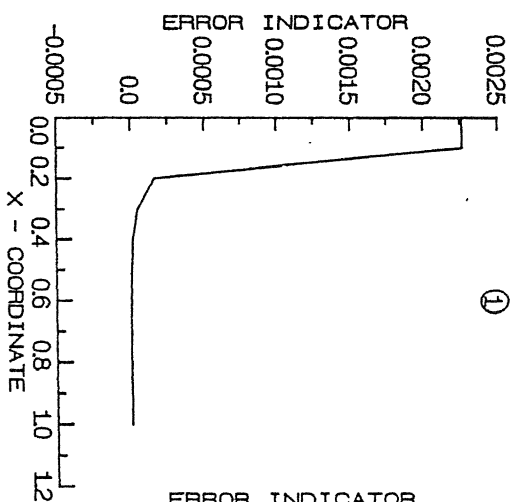
Graph ③

| Rdp | Nem | Nod |
|-----|-----|-----|
| 3 | 12 | 13 |

Graph ④

| Rdp | Nem | Nod |
|-----|-----|-----|
| 4 | 16 | 17 |

Fig. 5.12. TEST PROBLEM : BEHAVIOUR OF ERRORS (h VERSION ; $V = 5.0$)
($t = 0.00005$)



Graph ①

| Rdp | Nam | Nod |
|-----|-----|-----|
| 1 | 10 | 11 |
| | | |

Graph ②

| Rdp | Nam | Nod |
|-----|-----|-----|
| 2 | 10 | 11 |
| | | |

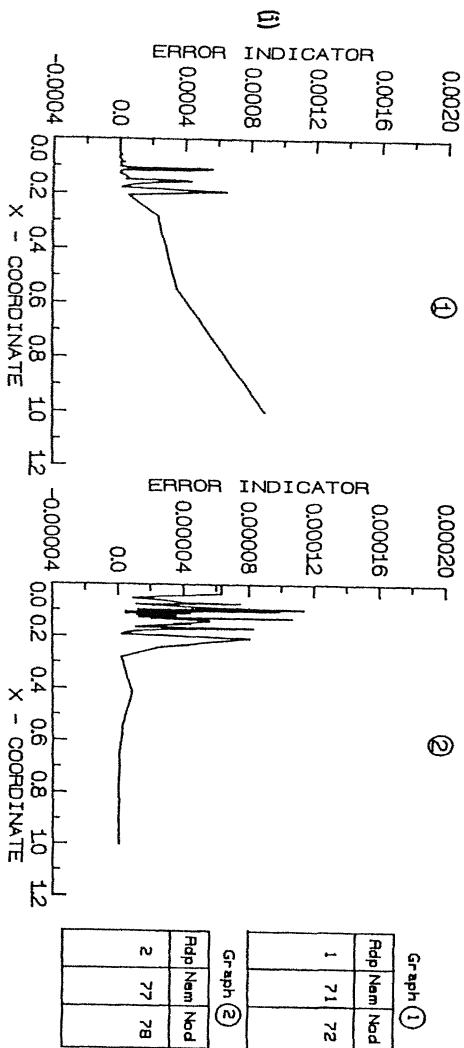
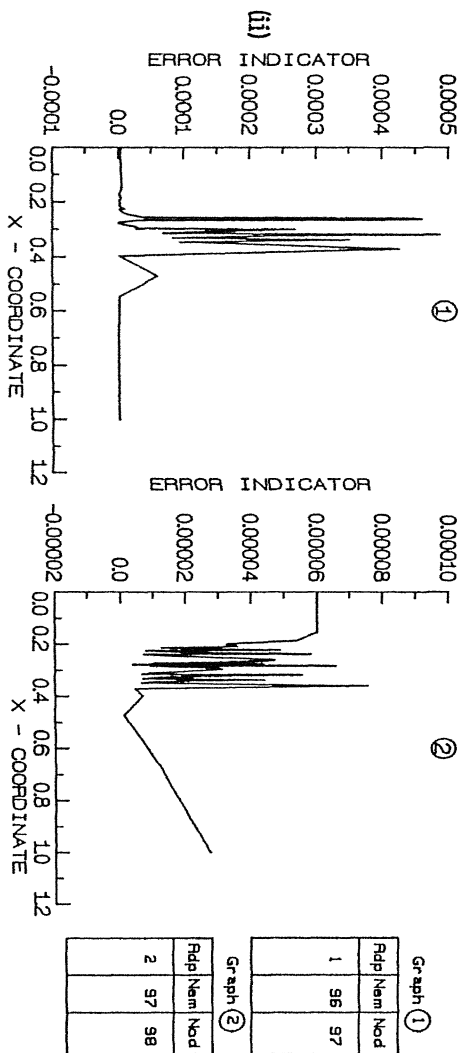
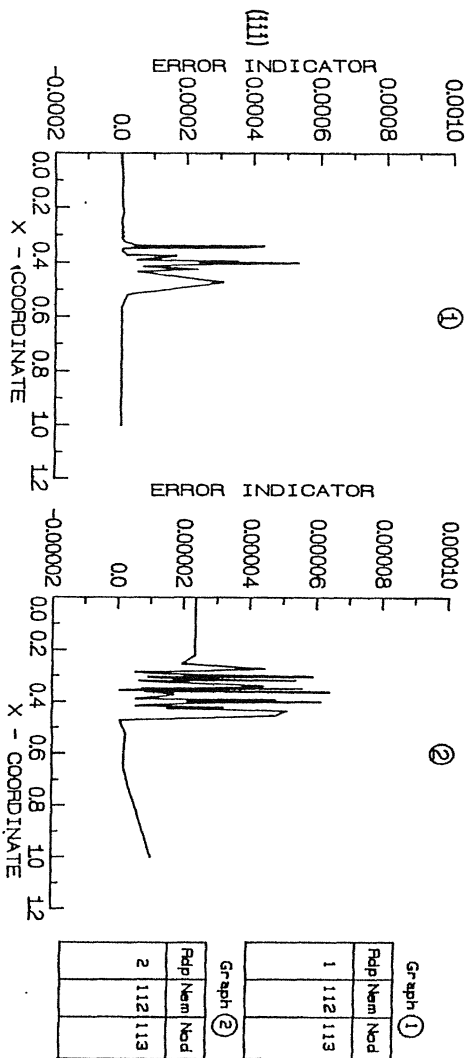
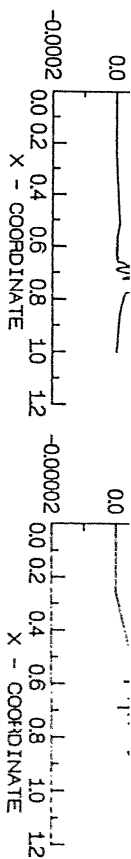


Fig 513. TEST PROBLEM : BEHAVIOUR OF ERRORS In VERSION ; V= 5.0
 ((i). $t = 0.0123$, (ii). $t = 0.02775$, (iii). $t = 0.03560$)



| | | |
|--|--|--|
| | | |
|--|--|--|

Graph (1)

| Idp | Nem | Ned |
|-----|-----|-----|
| 1 | 152 | 153 |

Graph (2)

| Idp | Nem | Ned |
|-----|-----|-----|
| 2 | 159 | 160 |

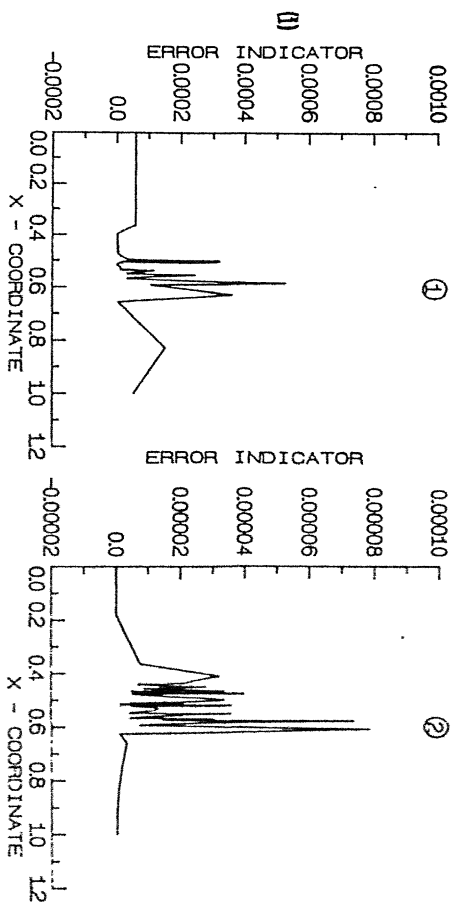
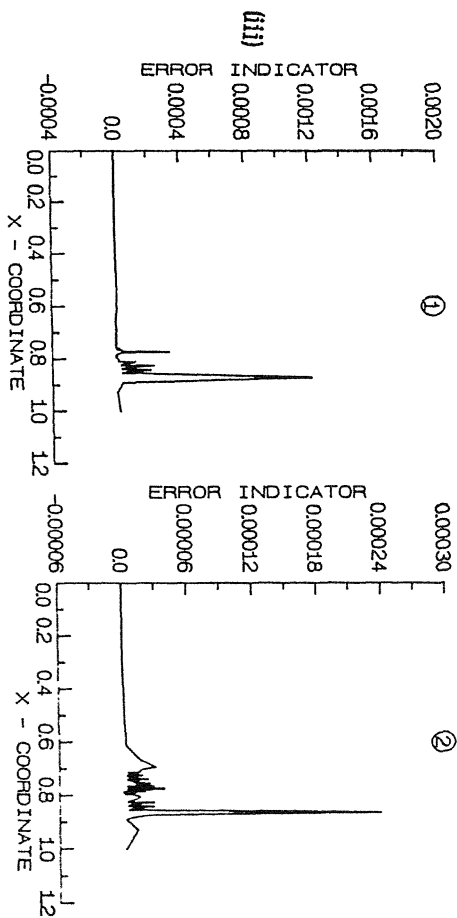


Fig. 5.14. TEST PROBLEM : BEHAVIOUR OF ERRORS IN VERSION : V= 5.0
(i) $t = 0.0514$, (ii) $t = 0.0687$, (iii) $t = 0.07885$



| Rdp | Nem | Nod |
|-----|-----|-----|
| 1 | 215 | 216 |
| | | |

Graph ②

| Rdp | Nem | Nod |
|-----|-----|-----|
| 2 | 214 | 215 |
| | | |

①

②

| Rdp | Nem | Nod |
|-----|-----|-----|
| 1 | 164 | 165 |
| | | |

Graph ①

| Rdp | Nem | Nod |
|-----|-----|-----|
| 2 | 170 | 171 |
| | | |

Graph ②

The exact and the computed solutions of the test problem are compared at various time levels in Fig. 5.15 for the case $V = 0$. It is observed that both the solutions match excellently at each time, during the front movement indicating that the refinement process works very well and is very accurate. The grid distributions for various time instants are shown in Fig. 5.16. It is evident that the α -strategy provides dense refinement in the front zone and that the grid distribution faithfully follows the front movement with respect to time. The other features are similar to those observed for the non-zero velocity case. The spatial distribution of local error for the non-convective situation ($V = 0$) are shown in the Figs. 5.17-5.19. Here also, the band of sharp error peaks moves with time, in response to a similar movement in the front location, as observed in the convective case. The mean and the maximum of the residuals as well as the error indicators have been shown in the Fig. 5.20. These also have similar characteristics as discussed in the convective problem. Also, the figure shows that the value of index α as expected, is nearly two during the uniform front motion. Further, the results for the case $V = 0.0$ are presented in the table 5.1.

5.7.2. Application of α -strategy to the flame propagation problem :

For the flame propagation study, some results have been predicted using uniform meshes in addition to the adaptive grids of the h-version α -strategy. This is in view of the fact that analytical solutions are not available for the non-linear flame propagation problem and uniform mesh solutions (on sufficiently refined grids) can be used for the sake of comparison. Moreover,

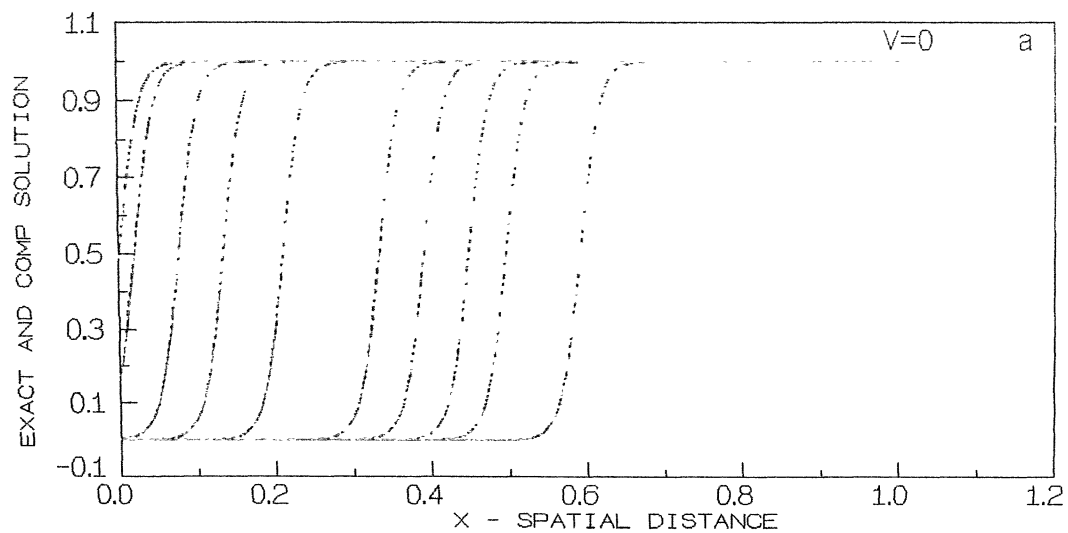


Fig.5.15 TEST PROBLEM (h version) : SOLUTION PROFILE

| Solution | Line_type |
|------------|-----------|
| h version | ----- |
| Exact soln | ---- |

Fig. (a)

| Adp | Time | Eles | Nods |
|-----|-----------|------|------|
| 1 | 0 .000050 | 15 | 16 |
| 2 | 0 .001950 | 21 | 22 |
| 3 | 0 .007650 | 50 | 51 |
| 4 | 0 .013350 | 126 | 127 |
| 5 | 0 .021000 | 142 | 143 |
| 6 | 0 .033350 | 152 | 153 |
| 7 | 0 .039050 | 162 | 163 |
| 8 | 0 .044800 | 171 | 172 |
| 9 | 0 .049550 | 175 | 176 |
| 10 | 0 .059050 | 217 | 218 |

Fig. (b)

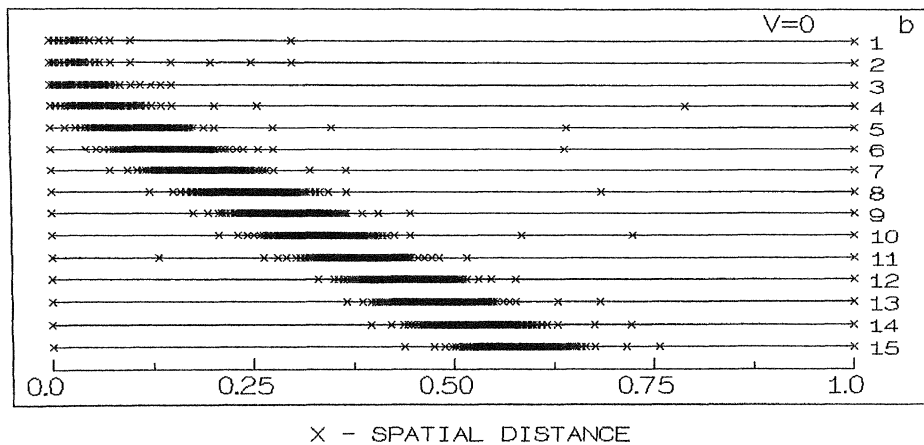


Fig.5.16 TEST PROBLEM (h version) : ADAPTED MESH AT VARIOUS t

| Adp | Time | Eles | Nods |
|-----|-----------|------|------|
| 1 | 0 .000050 | 15 | 16 |
| 2 | 0 .001950 | 21 | 22 |
| 3 | 0 .003850 | 35 | 36 |
| 4 | 0 .006700 | 50 | 51 |
| 5 | 0 .010500 | 96 | 97 |
| 6 | 0 .015250 | 135 | 136 |
| 7 | 0 .020050 | 140 | 141 |
| 8 | 0 .024800 | 161 | 162 |
| 9 | 0 .029550 | 160 | 161 |
| 10 | 0 .034300 | 151 | 152 |
| 11 | 0 .039050 | 162 | 163 |
| 12 | 0 .044800 | 171 | 172 |
| 13 | 0 .048600 | 178 | 179 |
| 14 | 0 .053350 | 195 | 196 |
| 15 | 0 .059050 | 217 | 218 |

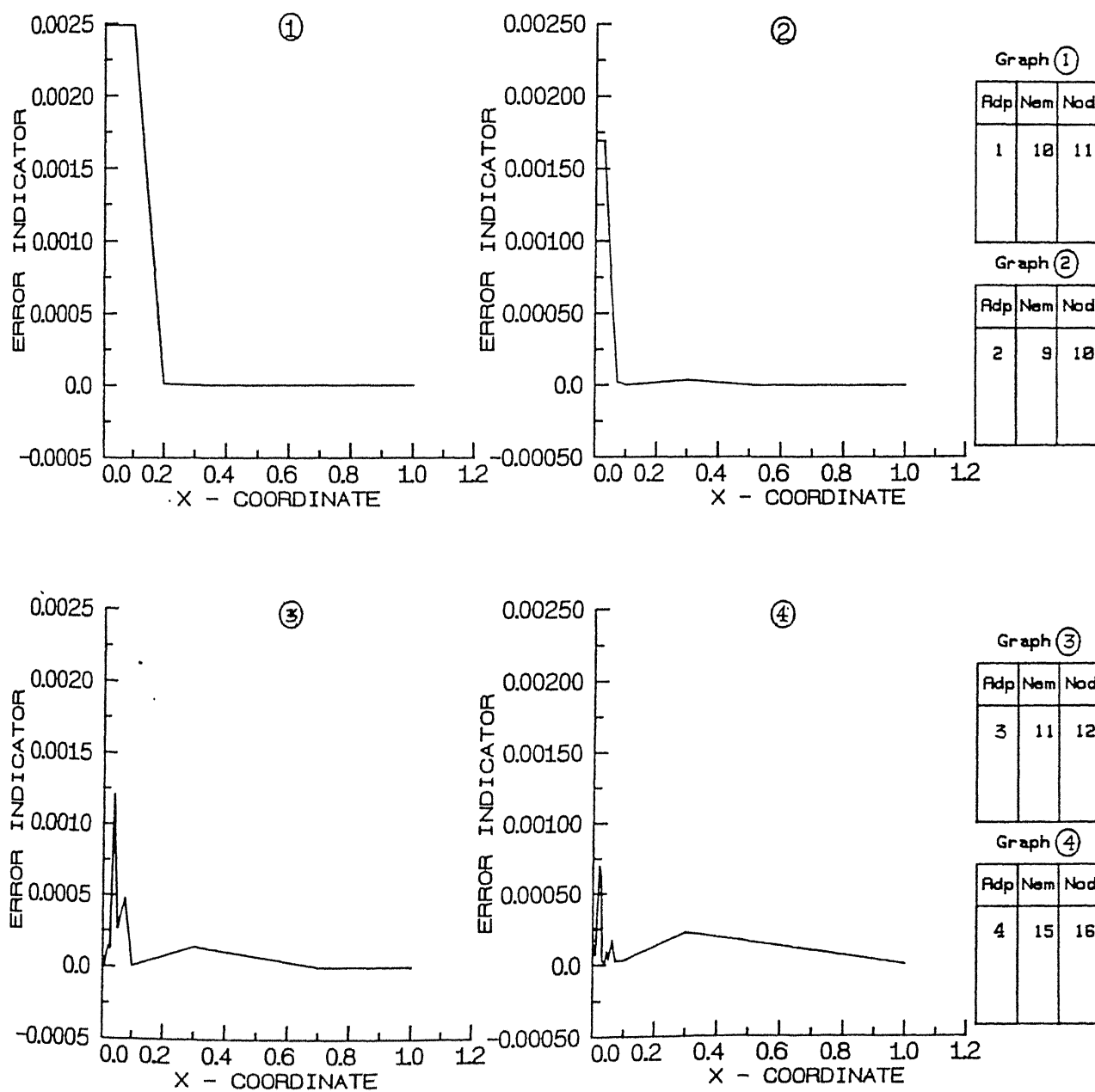


Fig. 5.17. TEST PROBLEM : BEHAVIOUR OF ERRORS (h VERSION ; $V = 0.0$)
($t = 0.00005$)

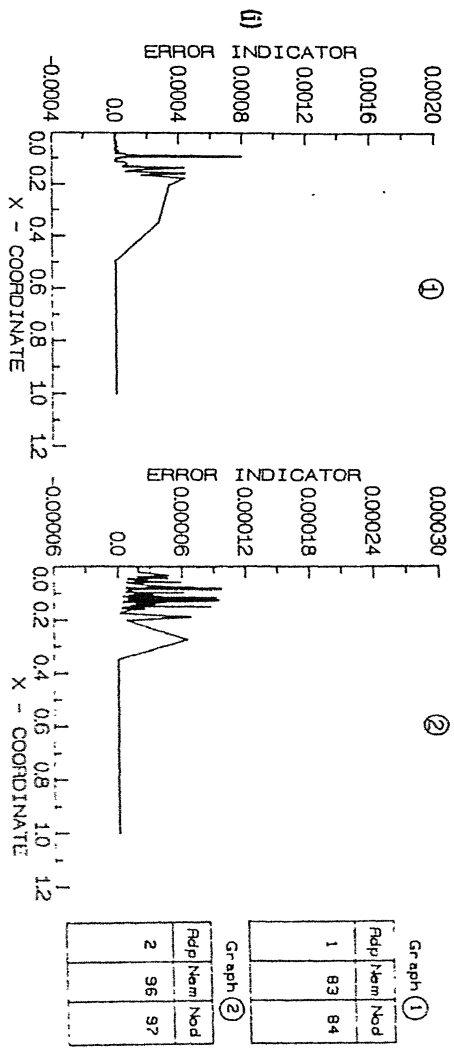
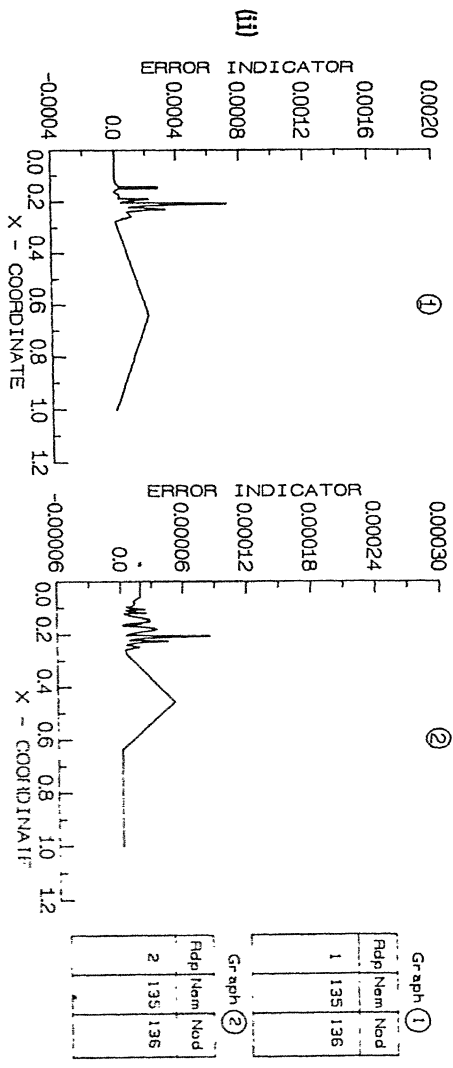
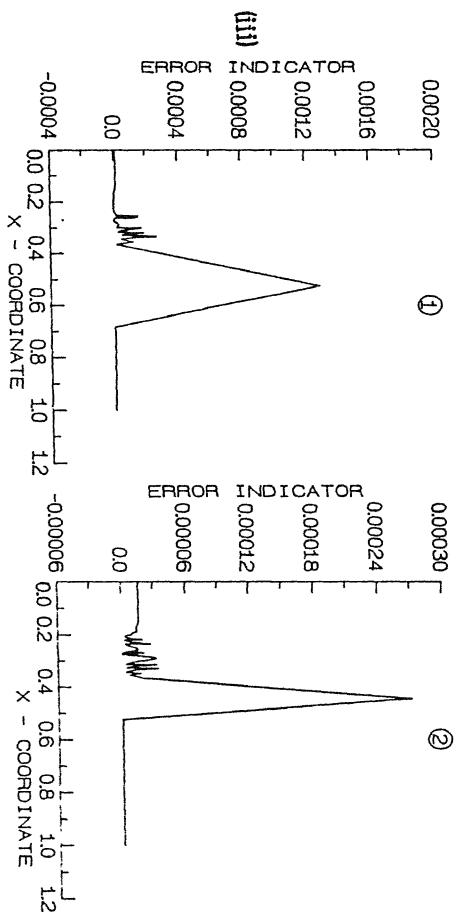


Fig. 518. TEST PROBLEM : BEHAVIOUR OF ERRORS (n VERSION : V= 0.0)
 (i) $t = 0.01045$, (ii) $t = 0.01615$ (iii) $t = 0.02725$



Graph ①

| Fddp | Nem | Nod |
|------|-----|-----|
| 1 | 168 | 169 |

Graph ②

| Fddp | Nem | Nod |
|------|-----|-----|
| 2 | 166 | 167 |

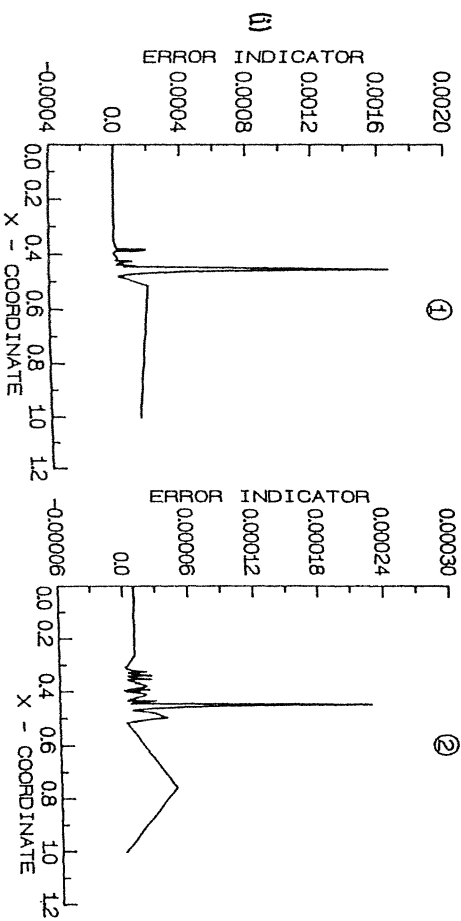
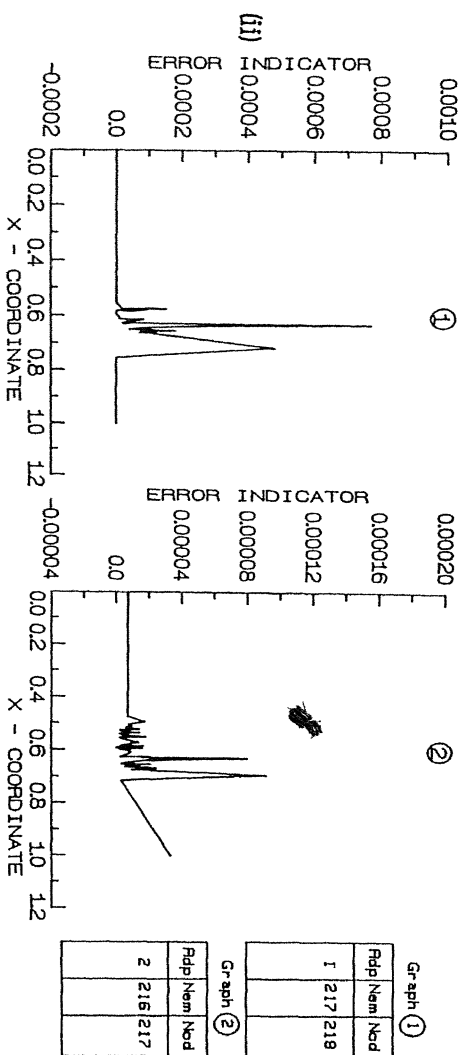
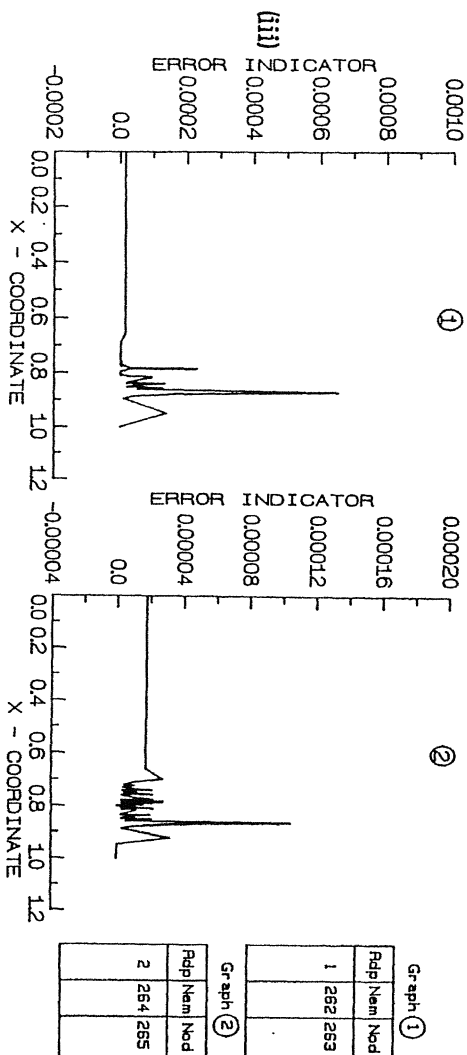


Fig. 549. TEST PROBLEM : BEHAVIOUR OF ERRORS (n VERSION : V= 0.0)
 (At $t = 0.03955$, $t = 0.05915$ and $t = 0.07995$)

| Graph ① | | |
|---------|-----|-----|
| Rdp | Nam | Nod |
| 1 | 162 | 163 |

| Graph ② | | |
|---------|-----|-----|
| Rdp | Nam | Nod |
| 2 | 164 | 165 |



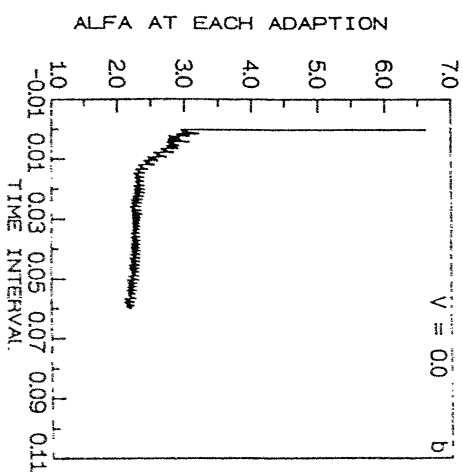
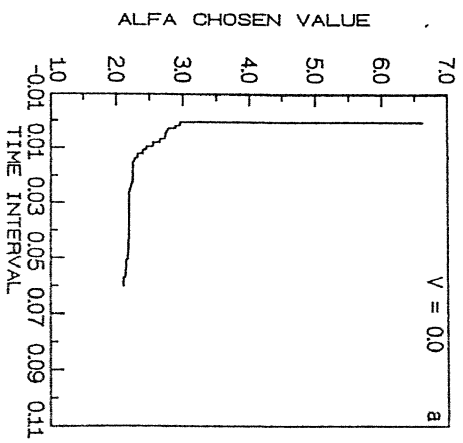
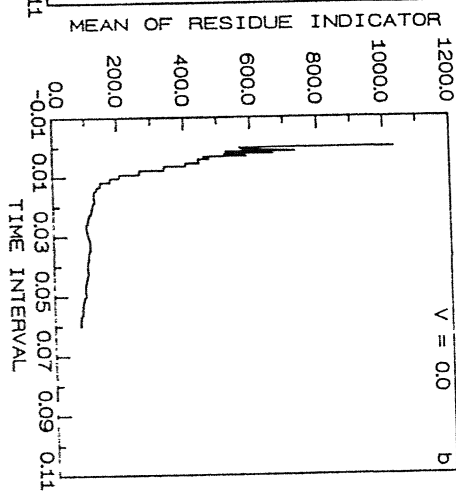
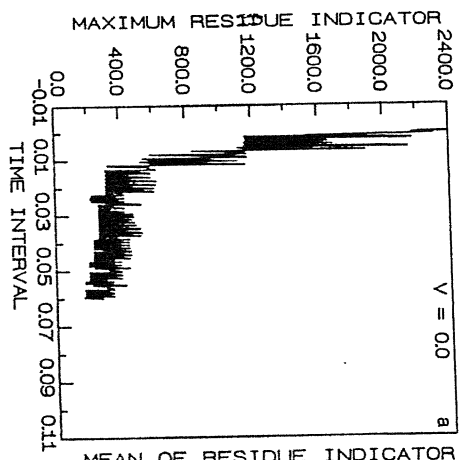
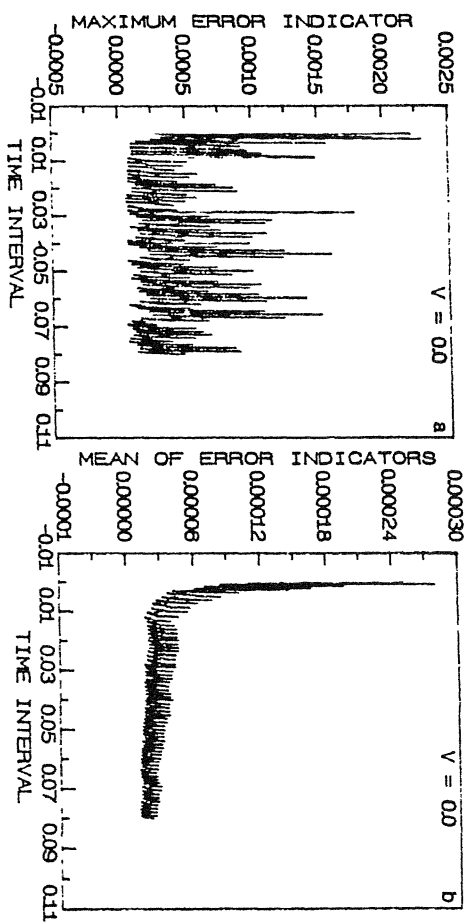


Fig. 5.20. TEST PROBLEM : TIME EVOLUTION OF ALFA VALUES,
RESIDUES AND ERRORS (n VERSION : V= 0.0)



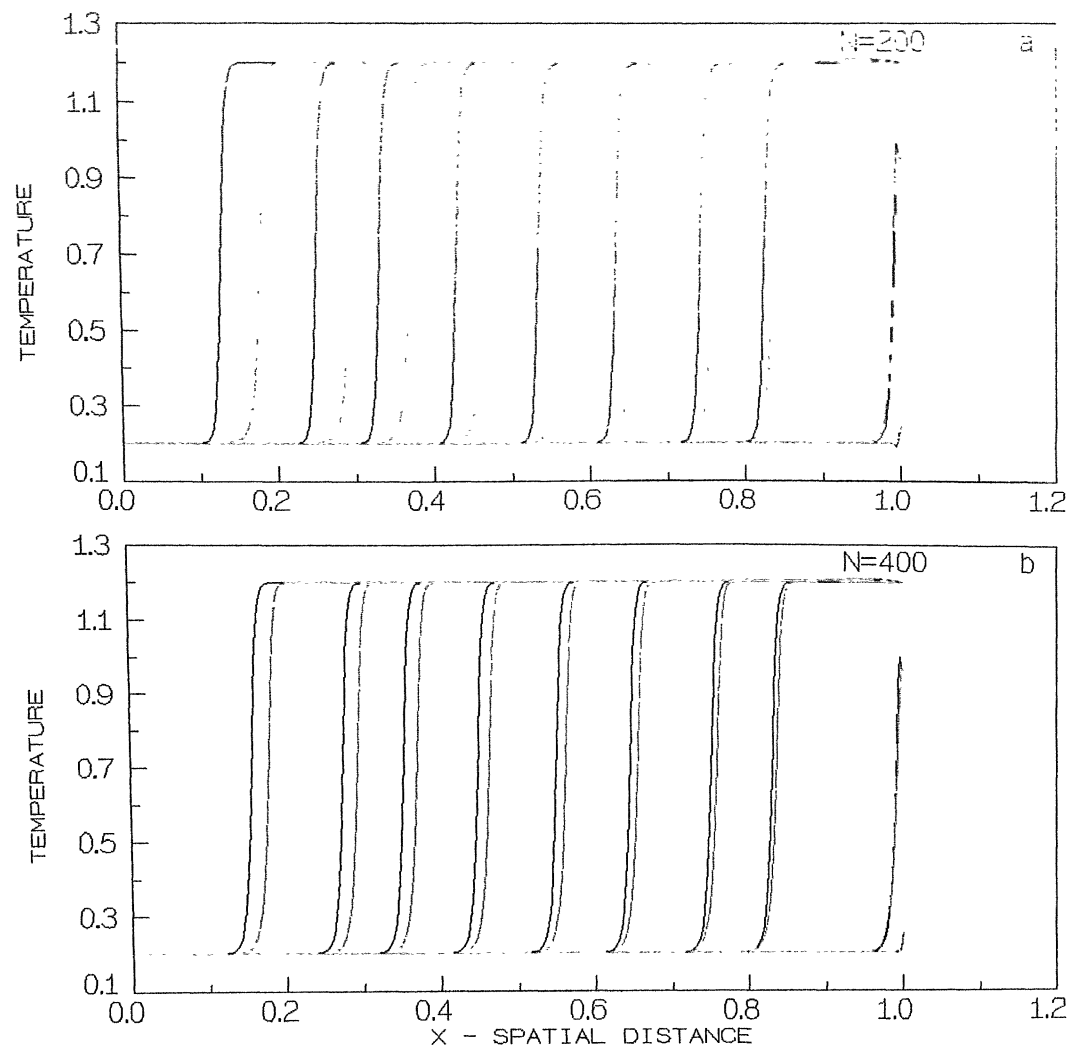
| Time | Adp | Elements | | | N | Apos.err | Grd.err | α -used | α -adp |
|---------|-----|----------|-----|-----|-----|----------|---------|----------------|---------------|
| | | Rem | Ref | Stf | | | | | |
| 0.00005 | 1 | 0 | 0 | 0 | 10 | 0.0500 | 0.1535 | 0.0000 | 6.6162 |
| | 2 | 4 | 1 | 4 | 9 | 0.0499 | 0.1535 | 6.6162 | 4.0771 |
| | 3 | 2 | 2 | 6 | 11 | 0.0499 | 0.1535 | 4.0771 | 3.2666 |
| | 4 | 1 | 3 | 8 | 15 | 0.0500 | 0.1535 | 3.2666 | 2.9553 |
| 0.00035 | 1 | 0 | 0 | 0 | 15 | 0.0509 | 0.1500 | 2.9553 | 2.9639 |
| 0.00090 | 1 | 0 | 0 | 0 | 15 | 0.0585 | 0.1534 | 2.9553 | 3.0347 |
| 0.00145 | 1 | 0 | 0 | 0 | 15 | 0.0772 | 0.1342 | 2.9553 | 3.2227 |
| | 2 | 0 | 4 | 10 | 21 | 0.0500 | 0.1342 | 2.9553 | 2.8894 |
| 0.00780 | 1 | 0 | 0 | 0 | 50 | 0.0769 | 0.0741 | 2.6563 | 2.8345 |
| | 2 | 8 | 16 | 38 | 64 | 0.0500 | 0.0741 | 2.6563 | 2.5500 |
| 0.01045 | 1 | 0 | 0 | 0 | 83 | 0.0766 | 0.0508 | 2.4536 | 2.6123 |
| | 2 | 17 | 20 | 50 | 96 | 0.0501 | 0.0508 | 2.4536 | 2.3999 |
| 0.01335 | 1 | 0 | 0 | 0 | 116 | 0.0770 | 0.0356 | 2.3157 | 2.4585 |
| | 2 | 20 | 30 | 60 | 126 | 0.0500 | 0.0356 | 2.3157 | 2.2693 |
| 0.01615 | 1 | 0 | 0 | 0 | 135 | 0.0761 | 0.0340 | 2.2498 | 2.3938 |
| | 2 | 21 | 21 | 42 | 135 | 0.0500 | 0.0340 | 2.2498 | 2.2546 |
| 0.01885 | 1 | 0 | 0 | 0 | 133 | 0.0768 | 0.0325 | 2.2498 | 2.4048 |
| | 2 | 21 | 28 | 56 | 140 | 0.0501 | 0.0325 | 2.2498 | 2.2461 |
| 0.02010 | 1 | 0 | 0 | 0 | 140 | 0.0772 | 0.0321 | 2.2461 | 2.4085 |
| | 2 | 23 | 25 | 50 | 142 | 0.0500 | 0.0321 | 2.2461 | 2.2473 |
| 0.02945 | 1 | 0 | 0 | 0 | 163 | 0.0762 | 0.0290 | 2.1899 | 2.3547 |
| | 2 | 20 | 17 | 34 | 160 | 0.0499 | 0.0290 | 2.1899 | 2.2144 |
| 0.03610 | 1 | 0 | 0 | 0 | 157 | 0.0772 | 0.0284 | 2.1899 | 2.3425 |
| | 2 | 25 | 26 | 52 | 158 | 0.0500 | 0.0284 | 2.1899 | 2.2022 |
| 0.04900 | 1 | 0 | 0 | 0 | 178 | 0.0769 | 0.0258 | 2.1741 | 2.3181 |
| | 2 | 31 | 28 | 56 | 175 | 0.0500 | 0.0258 | 2.1741 | 2.1704 |
| 0.06155 | 1 | 0 | 0 | 0 | 216 | 0.0761 | 0.0199 | 2.1057 | 2.2485 |
| | 2 | 44 | 54 | 108 | 226 | 0.0500 | 0.0199 | 2.1057 | 2.0960 |
| 0.06790 | 1 | 0 | 0 | 0 | 247 | 0.0762 | 0.0189 | 2.0862 | 2.2363 |
| | 2 | 43 | 44 | 88 | 248 | 0.0500 | 0.0189 | 2.0862 | 2.0825 |
| 0.07515 | 1 | 0 | 0 | 0 | 267 | 0.0502 | 0.0176 | 2.0587 | 2.0599 |
| 0.07630 | 1 | 0 | 0 | 0 | 267 | 0.0770 | 0.0176 | 2.0587 | 2.1973 |
| | 2 | 66 | 68 | 136 | 269 | 0.0500 | 0.0176 | 2.0587 | 2.0581 |
| 0.07745 | 1 | 0 | 0 | 0 | 269 | 0.0772 | 0.0180 | 2.0581 | 2.1948 |
| | 2 | 68 | 66 | 132 | 267 | 0.0500 | 0.0180 | 2.0581 | 2.0624 |
| 0.07860 | 1 | 0 | 0 | 0 | 267 | 0.0770 | 0.0181 | 2.0581 | 2.1899 |
| | 2 | 75 | 70 | 140 | 262 | 0.0500 | 0.0181 | 2.0581 | 2.0703 |
| 0.07905 | 1 | 0 | 0 | 0 | 262 | 0.0569 | 0.0200 | 2.0581 | 2.0960 |

Table. 5.1. Test problem results for
h version based α strategy ($V = 0.0$)

they are useful for analyzing the effectiveness of grid adaption. In the ensuing sections, the behavior of the solution profiles, grid distributions and various error measures are presented and discussed.

(a). Solution profiles and grid distributions :

In Fig. 5.21, the solution profiles of α -strategy are compared with those obtained using uniform meshes. The upwinding finite element procedure has been used for here to obtain the numerical solutions. During the ignition phase, even a coarse mesh (of just 18 elements) used in conjunction with the α -strategy provides as accurate a result as the uniform mesh of 400 elements for $V = 5.0$. When the flame front moves at a constant speed, the predictions over the uniform mesh are not accurate enough. For a 200 element uniform mesh, the results are very inaccurate, especially in estimating the correct propagation speed of the front. The shape of the front is predicted reasonably well. The results of a 400 element uniform mesh are closer to those obtained by the α -strategy; even then small inaccuracies still remain in the predictions of the uniform mesh. This is in view of the fact that grid point locations are optimal at each time instant in the error based α -strategy, while in the uniform mesh many elements are located in regions where they are not needed. More discussion on the high accuracy of predictions for the α -strategy is provided subsequently in connection with Fig. 5.23. Thus the adaptive strategy provides a superior performance in terms of good solution accuracy, even with less number of elements. It must, however, be mentioned here that the number of elements increases with time for



| Solution | Line_type |
|-----------|-----------|
| uniform | ———— |
| h version | ----- |

h version mesh

| Adp | Time | Eles | Nods |
|-----|-----------|------|------|
| 1 | 0 .000010 | 18 | 19 |
| 2 | 0 .000150 | 77 | 78 |
| 3 | 0 .001250 | 211 | 212 |
| 4 | 0 .001820 | 206 | 207 |
| 5 | 0 .002570 | 275 | 276 |
| 6 | 0 .003250 | 277 | 278 |
| 7 | 0 .003970 | 295 | 296 |
| 8 | 0 .004650 | 321 | 322 |
| 9 | 0 .005210 | 389 | 390 |
| 10 | 0 .006050 | 460 | 461 |

Fig. 5.21 FLAME PROBLEM (Uniform & h Version Mesh : V=5.0) : SOLUTION PROFILE

the α -strategy till the error estimate asymptotically stabilizes; the total number of elements required to meet the *a posteriori* error criterion after the stabilization of α , depends upon the values of the error limits (δ_1 and δ_2). With the proper choice of these limits, quite accurate results can be obtained with moderate computational effort.

In order to test the general applicability of α -strategy, an extreme case of $V = 200.0$ has been considered in Fig. 5.22. Due to the very high flow velocity, it is expected that the front motion will be completely arrested in this situation. Indeed for large time, steady profiles of steep variation in the temperature and density are observed, adjacent to $x = 1.0$, indicating attached flame at this boundary.

The solution profiles obtained by the α -strategy are compared with those predicted by the conventional h-version method for the no flow situation of $V = 0$ in Fig. 5.23. Excellent agreement is seen in Fig. 5.23 and any minor deviations present are solely attributable to the inaccuracies of the uniform mesh solutions. The number of elements is somewhat higher for the h-version method in the beginning. However, this trend reverses with α -strategy requiring more elements, for later times. Initially, on a coarse mesh, the value of α is high and it stabilizes slowly with time. Due to this, the α -strategy employs less elements (or larger average step size) to meet the error criterion in the beginning. However, when the value of α is stabilized, the α -strategy needs more elements, due to the fact that it uses two error limits ($\delta_1 = 0.50$, and $\delta_2 = 0.75$), with the upper limit δ_2 coinciding with the error tolerance of the h-version method (δ_2

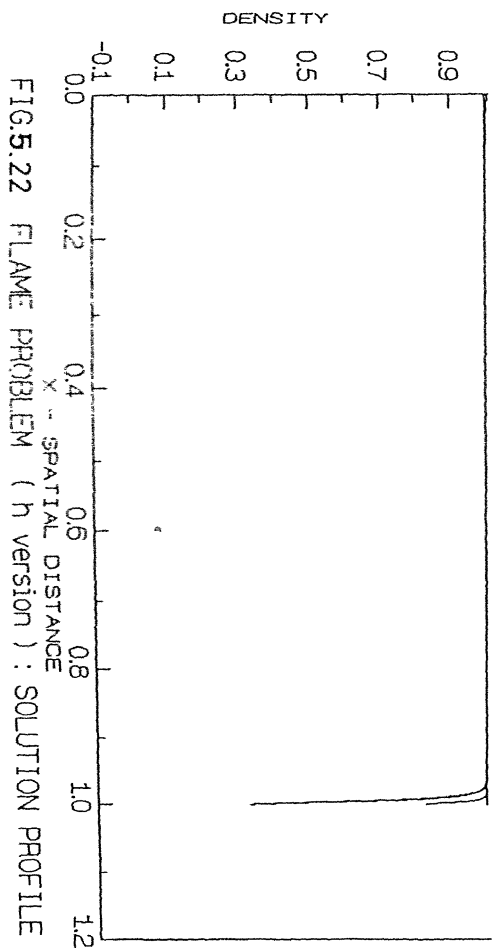
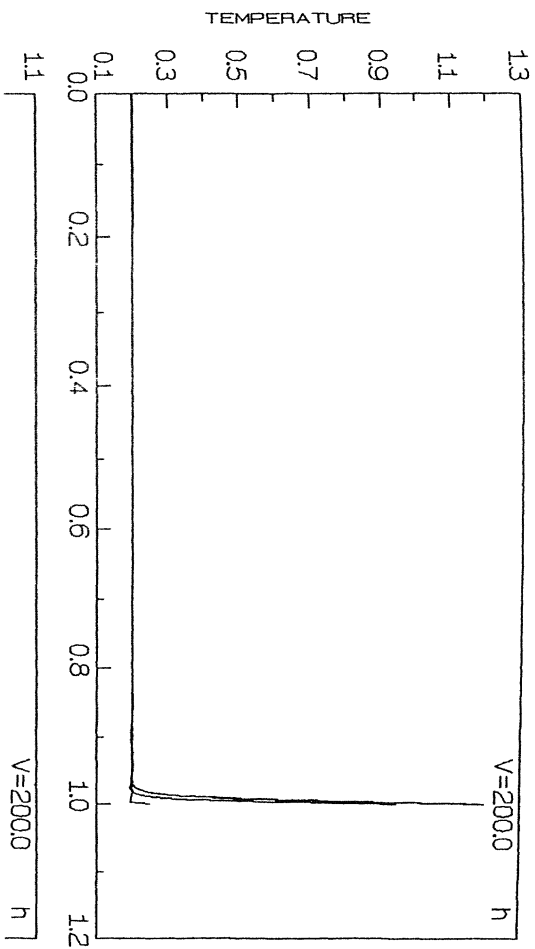


FIG.5.22 FLAME PROBLEM (h version) : SOLUTION PROFILE



| ADP | Time | Eles | Nodos |
|-----|-----------|------|-------|
| 1 | 0 .000010 | 14 | 15 |
| 2 | 0 .000150 | 32 | 33 |
| 3 | 0 .000450 | 79 | 80 |
| 4 | 0 .000850 | 125 | 126 |
| 5 | 0 .001150 | 144 | 145 |
| 6 | 0 .001450 | 150 | 151 |
| 7 | 0 .001900 | 202 | 203 |
| 8 | 0 .002450 | 200 | 201 |
| 9 | 0 .002950 | 229 | 230 |
| 10 | 0 .003450 | 241 | 242 |

= 0.075). Since it is required that a *posteriori* error should lie between δ_1 and δ_2 (with $\delta_1 < \delta_2$), the application of the error tolerance is more stringent in the α -strategy, which explains the larger number of elements during the later stage of front propagation.

In Figs. 5.24 and 5.25, the grid distributions are shown for the velocity values of $V = 0.0$ (see Table 5.2) and $V = 5.0$ at various times. All the features of the adaptive meshes such as dense refinement in the front zone, movement of the grid points along with the front etc., are clearly observed. Indeed the fact that the value of the exponent α is not initially stabilized does not seem to affect the nature of mesh refinement. Thus there is a strong case for using the α -strategy in nonlinear problems for which theoretical estimates of error are not available and the *a posteriori* improvement of error estimation may be the only suitable option. Comparing the convective and non-convective situations it can be said that since the flow velocity direction is opposite to the front propagation, the effective flame speed is reduced when the flow is present; therefore less number of grid points are sufficient to meet the error criterion. This feature is clearly evident for an extremely high flow velocity $V = 200.0$ as seen from the table provided with Fig. 5.22. It appears that one of the main reasons why a large number of grid points are required to study flame propagation is the high speed of the flame front, due to which a zone thicker than the flame front has to be refined. Therefore reduction in the effective flame speed leads to reduction in the number of elements.

The grid evolution with time and adaption level, is shown in

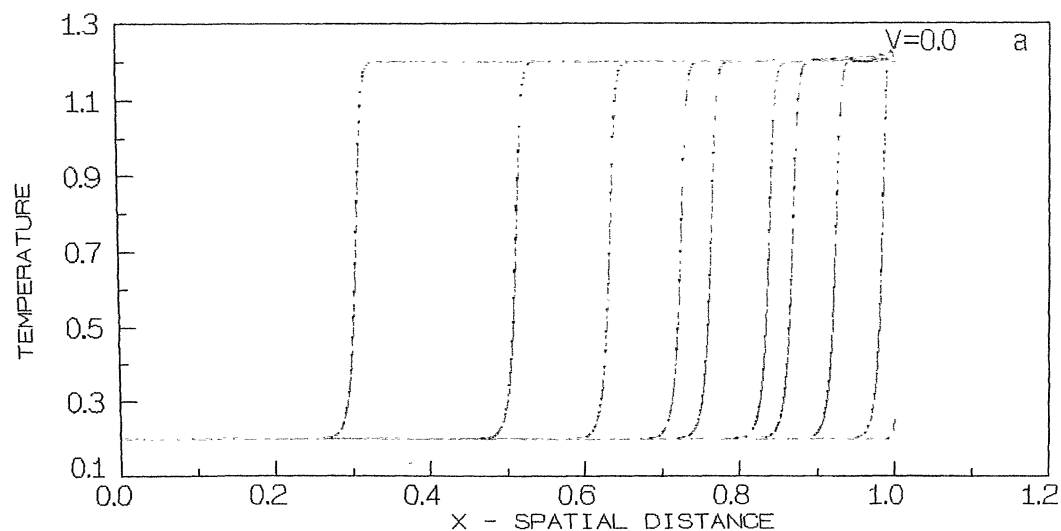


Fig. 5.23 FLAME PROBLEM (h version) : SOLUTION PROFILE

| Solution | Line_type |
|-----------|-----------|
| strategy | ----- |
| h version | |

Fig. (a)

| Adp | Time | α -st | h |
|-----|----------|--------------|-----|
| 1 | 0.000010 | 15 | 13 |
| 2 | 0.000200 | 220 | 315 |
| 3 | 0.000600 | 287 | 369 |
| 4 | 0.000990 | 319 | 350 |
| 5 | 0.001200 | 298 | 357 |
| 6 | 0.001730 | 307 | 323 |
| 7 | 0.002000 | 346 | 342 |
| 8 | 0.002650 | 376 | 355 |
| 9 | 0.003500 | 405 | 330 |
| 10 | 0.004100 | 460 | 353 |

Fig. (b)

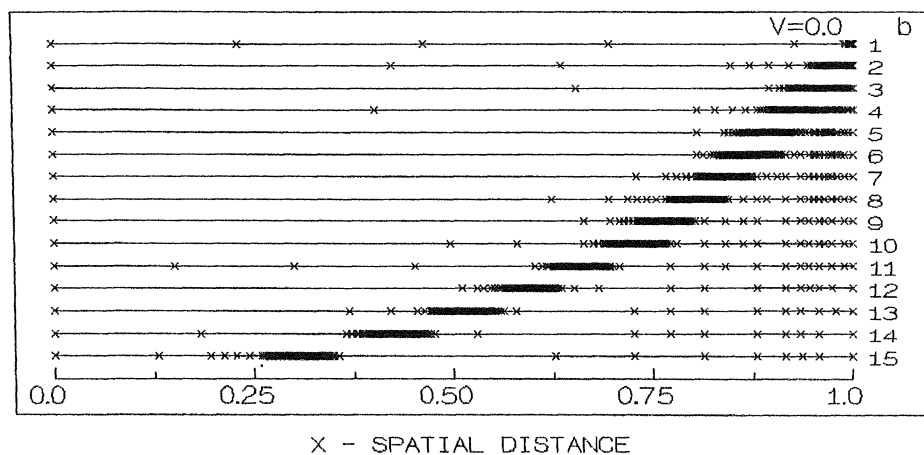
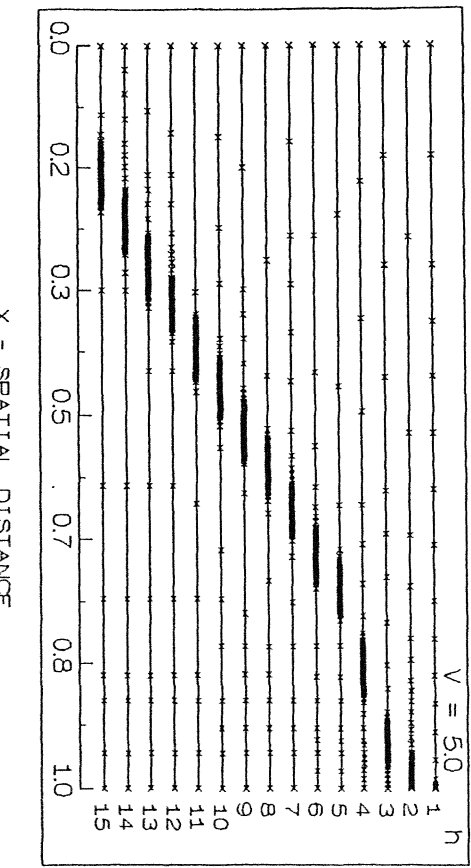


Fig. 5.24 FLAME PROBLEM (h version) : ADAPTED MESH AT VARIOUS t

| Adp | Time | α -st | Nods |
|-----|----------|--------------|------|
| 1 | 0.000010 | 15 | 16 |
| 2 | 0.000200 | 220 | 221 |
| 3 | 0.000400 | 258 | 259 |
| 4 | 0.000600 | 287 | 288 |
| 5 | 0.000800 | 313 | 314 |
| 6 | 0.000990 | 319 | 320 |
| 7 | 0.001200 | 298 | 299 |
| 8 | 0.001400 | 282 | 283 |
| 9 | 0.001730 | 307 | 308 |
| 10 | 0.002000 | 346 | 347 |
| 11 | 0.002450 | 354 | 355 |
| 12 | 0.002900 | 375 | 376 |
| 13 | 0.003500 | 405 | 406 |
| 14 | 0.004100 | 460 | 461 |
| 15 | 0.004950 | 493 | 494 |



| Adp | Time | Eles | Nods |
|-----|-----------|------|------|
| 1 | 0 .000010 | 18 | 19 |
| 2 | 0 .000150 | 77 | 78 |
| 3 | 0 .000500 | 96 | 97 |
| 4 | 0 .001250 | 211 | 212 |
| 5 | 0 .001990 | 195 | 196 |
| 6 | 0 .002370 | 248 | 249 |
| 7 | 0 .002730 | 234 | 235 |
| 8 | 0 .003170 | 263 | 264 |
| 9 | 0 .003570 | 297 | 298 |
| 10 | 0 .003970 | 295 | 296 |
| 11 | 0 .004330 | 321 | 322 |
| 12 | 0 .004770 | 297 | 298 |
| 13 | 0 .005130 | 367 | 368 |
| 14 | 0 .005610 | 387 | 388 |
| 15 | 0 .006050 | 460 | 461 |

FIG. 5.25 FLAME PROBLEM (h VERSION) ADAPTED MESH AT VARIOUS t
(V = 5.0)

| Time | Adp | Elements | | | N | Apos.err | Com.grd | α -used | α -adp |
|---------|-----|----------|-----|-----|-----|----------|---------|----------------|---------------|
| | | Rem | Ref | Stf | | | | | |
| 0.00001 | 1 | 0 | 0 | 0 | 10 | 0.04908 | 0.2078 | 0.0000 | 8.4770 |
| | 2 | 4 | 1 | 4 | 9 | 0.04907 | 0.4067 | 8.4770 | 4.9220 |
| | 3 | 2 | 1 | 4 | 10 | 0.04905 | 0.6596 | 4.9220 | 3.3200 |
| | 4 | 4 | 1 | 4 | 9 | 0.05054 | 0.7418 | 3.3200 | 2.5390 |
| | 5 | 2 | 3 | 8 | 12 | 0.04990 | 0.7484 | 2.5390 | 2.4220 |
| | 6 | 1 | 2 | 6 | 15 | 0.04912 | 0.7477 | 2.4220 | 2.2660 |
| 0.00002 | 1 | 0 | 0 | 0 | 15 | 0.22530 | 1.1851 | 2.2660 | 3.7500 |
| | 2 | 2 | 3 | 10 | 20 | 0.05037 | 1.1831 | 2.2660 | 2.5000 |
| 0.00003 | 1 | 0 | 0 | 0 | 20 | 0.19672 | 1.6113 | 2.2660 | 2.8520 |
| | 2 | 4 | 3 | 8 | 21 | 0.05074 | 1.6083 | 2.2660 | 2.4800 |
| 0.00010 | 1 | 0 | 0 | 0 | 51 | 0.07911 | 3.9135 | 2.2660 | 2.4020 |
| | 2 | 0 | 14 | 34 | 71 | 0.04944 | 3.9135 | 2.2660 | 2.3440 |
| 0.00016 | 1 | 0 | 0 | 0 | 135 | 0.09091 | 8.1376 | 2.2660 | 2.4120 |
| | 2 | 2 | 34 | 76 | 175 | 0.05028 | 8.1376 | 2.2660 | 2.4020 |
| 0.00021 | 1 | 0 | 0 | 0 | 220 | 0.08172 | 10.0421 | 2.2660 | 2.3930 |
| | 2 | 40 | 54 | 108 | 234 | 0.05048 | 10.0419 | 2.2660 | 2.2750 |
| 0.00027 | 1 | 0 | 0 | 0 | 237 | 0.07687 | 9.6903 | 2.2660 | 2.3730 |
| | 2 | 45 | 48 | 96 | 240 | 0.05018 | 9.6901 | 2.2660 | 2.2850 |
| 0.00034 | 1 | 0 | 0 | 0 | 253 | 0.07932 | 9.3440 | 2.2660 | 2.3830 |
| | 2 | 54 | 63 | 126 | 262 | 0.05034 | 9.3439 | 2.2660 | 2.2660 |
| 0.00042 | 1 | 0 | 0 | 0 | 258 | 0.08128 | 9.1093 | 2.2660 | 2.4020 |
| | 2 | 57 | 49 | 98 | 250 | 0.04974 | 9.1092 | 2.2660 | 2.2950 |
| 0.00048 | 1 | 0 | 0 | 0 | 263 | 0.07706 | 9.0007 | 2.2660 | 2.4020 |
| | 2 | 52 | 56 | 112 | 267 | 0.05077 | 9.0006 | 2.2660 | 2.2750 |
| 0.00067 | 1 | 0 | 0 | 0 | 286 | 0.07938 | 8.8373 | 2.2560 | 2.3830 |
| | 2 | 78 | 80 | 160 | 288 | 0.05061 | 8.8372 | 2.2560 | 2.2560 |
| 0.00088 | 1 | 0 | 0 | 0 | 315 | 0.07698 | 8.7821 | 2.2170 | 2.3240 |
| | 2 | 70 | 70 | 140 | 315 | 0.04909 | 8.7821 | 2.2170 | 2.2170 |
| 0.00148 | 1 | 0 | 0 | 0 | 294 | 0.07951 | 8.7590 | 2.2270 | 2.3440 |
| | 2 | 68 | 73 | 146 | 299 | 0.04991 | 8.7590 | 2.2270 | 2.2270 |
| 0.00217 | 1 | 0 | 0 | 0 | 377 | 0.08422 | 8.7570 | 2.1680 | 2.5000 |
| | 2 | 83 | 60 | 120 | 354 | 0.05075 | 8.7570 | 2.1680 | 2.3050 |
| 0.00240 | 1 | 0 | 0 | 0 | 345 | 0.07646 | 8.7574 | 2.1680 | 2.2950 |
| | 2 | 61 | 57 | 114 | 341 | 0.04965 | 8.7574 | 2.1680 | 2.1880 |
| 0.00247 | 1 | 0 | 0 | 0 | 354 | 0.07963 | 8.7570 | 2.1680 | 2.3050 |
| | 2 | 81 | 83 | 166 | 356 | 0.05060 | 8.7570 | 2.1680 | 2.2070 |
| 0.00251 | 1 | 0 | 0 | 0 | 356 | 0.08330 | 8.7569 | 2.1680 | 2.3240 |
| | 2 | 86 | 97 | 194 | 367 | 0.04911 | 8.7569 | 2.1680 | 2.2070 |
| 0.00258 | 1 | 0 | 0 | 0 | 374 | 0.07803 | 8.7569 | 2.1680 | 2.2950 |
| | 2 | 94 | 94 | 188 | 374 | 0.04940 | 8.7569 | 2.1680 | 2.1780 |
| 0.00297 | 1 | 0 | 0 | 0 | 382 | 0.08010 | 8.7570 | 2.1680 | 2.3050 |
| | 2 | 13 | 106 | 212 | 375 | 0.04961 | 8.7569 | 2.1680 | 2.1880 |
| 0.00300 | 1 | 0 | 0 | 0 | 375 | 0.07038 | 8.7568 | 2.1680 | 2.2560 |

Table. 5.2. Flame propagation results for
h version based α strategy ($N = 0.0$)

Figs. 5.26a-b. The grid movement is observed to be uniform with respect to time during the equilibrium phase. For the numerical computations based on the α - strategy; large time step values have been taken as compared to the time steps used for the results of Chapter 4. Due to this, during the initial time steps, the prediction of regions for refinement is not very accurate. However, the situation improves as the marching in time takes place. Features such as the growth in the width of flame front during the ignition phase, constant width of the front, during stabilized phase etc. are observed, as expected. At every location in the burnt region, derefinement process takes place immediately after the flame front proceeds away. If derefinement is not as effective as refinement, element removal is slow in the burnt region, as can be seen from figures at some time instants. On the other hand, in the unburnt region, the mesh is always coarse due to the coarse initial mesh.

(b). Error behaviour :

In order to highlight the improvement in error estimation for the α -strategy, the variations of the *a posteriori* error and the exponent α with time have been presented in Fig 5.27 for two different uniform meshes having 200 and 400 elements, for the flow velocity value of $V = 5.0$. The *a posteriori* error rapidly increases in the beginning due to the highly nonlinear ignition process occurring near the $x = 1.0$ boundary. After these initial transients, the *a posteriori* error undergoes systematic bounded oscillations with time as discussed in Chapter 4. The oscillations in error can be attributed to the motion of the flame front

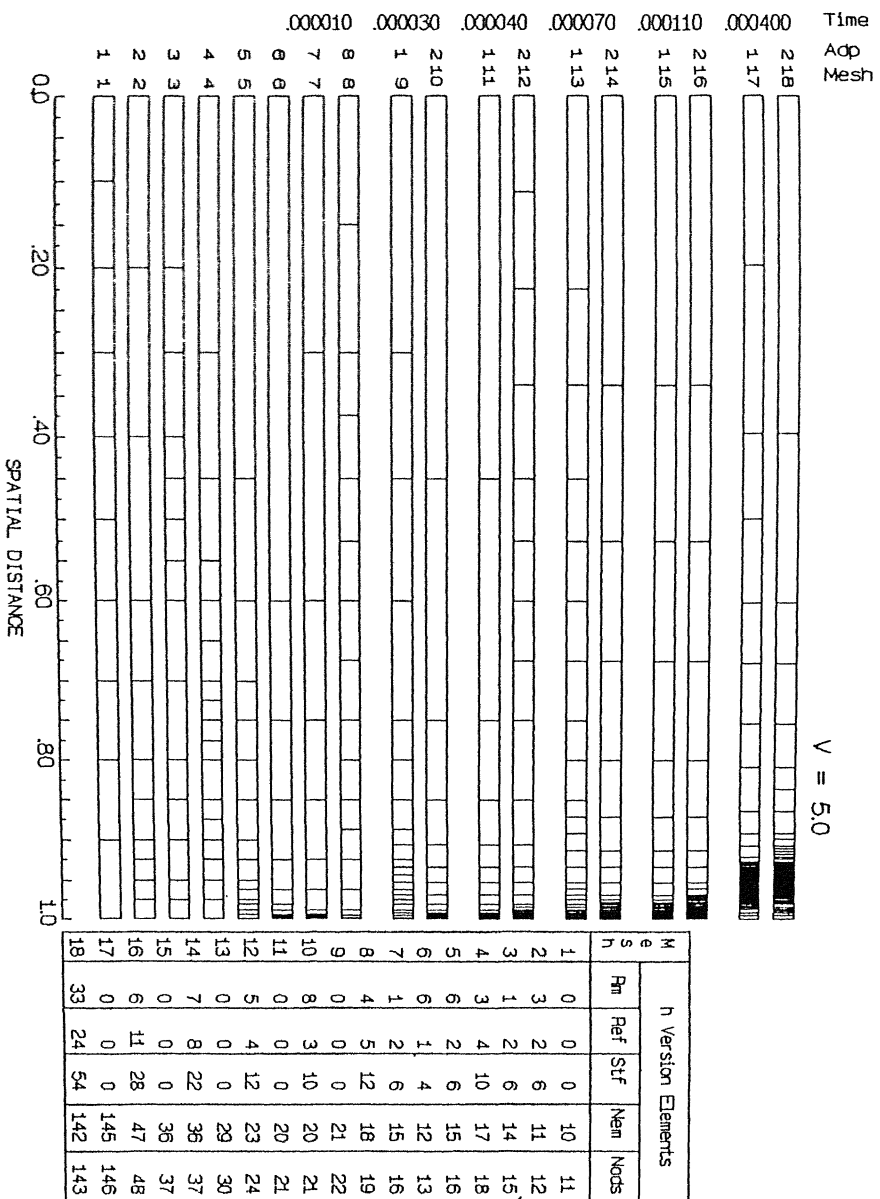


Fig. 5.26(a). FLAME PROBLEM : ADAPTED MESH AT VARIOUS TIME INSTANTS

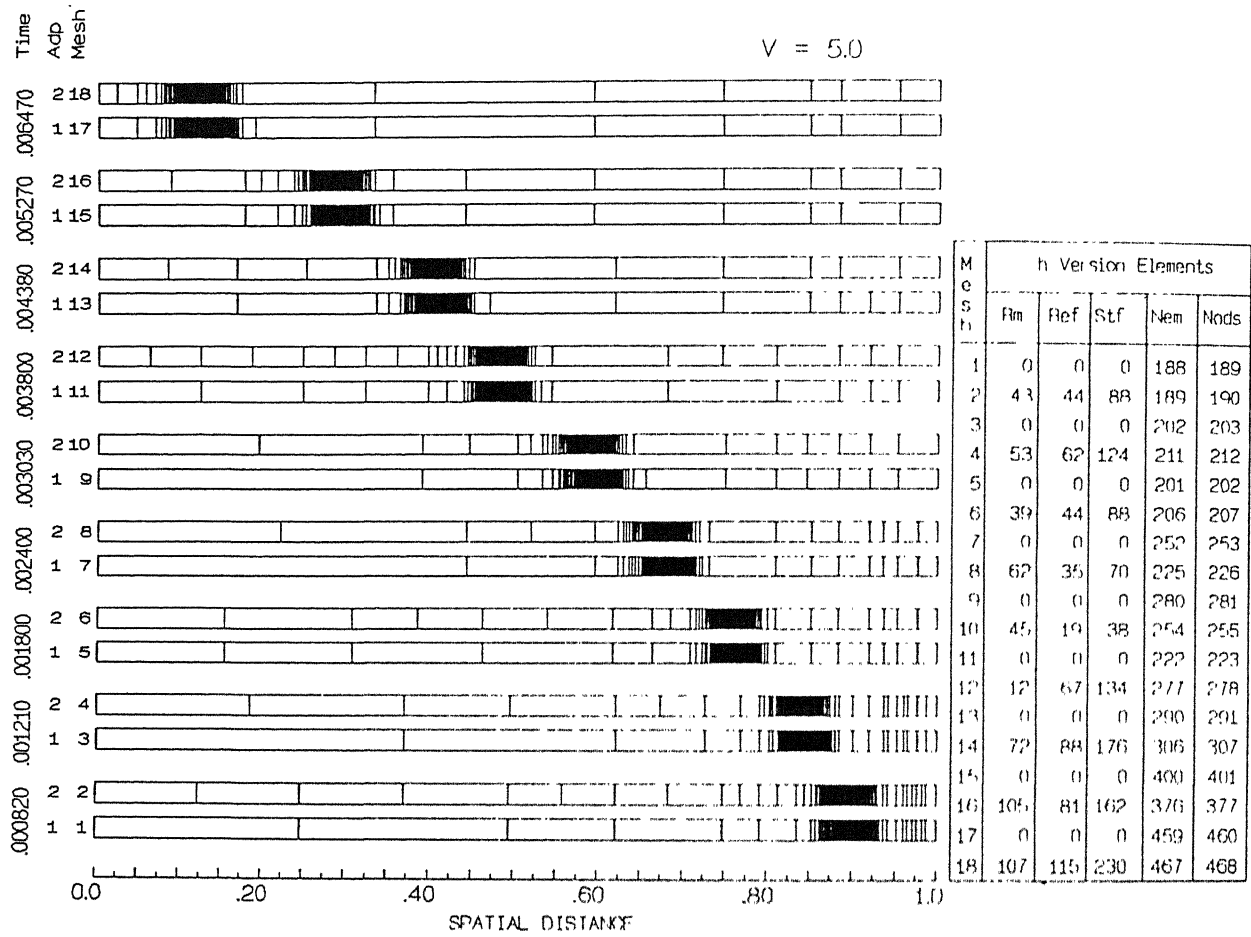


FIG. 5.26(b) h - ADAPTED MESH AT VARIOUS TIME INSTANTS (FLAME PROBLEM)

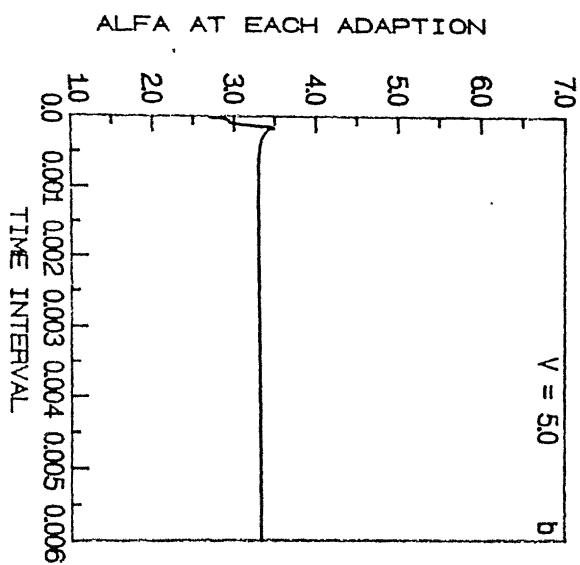
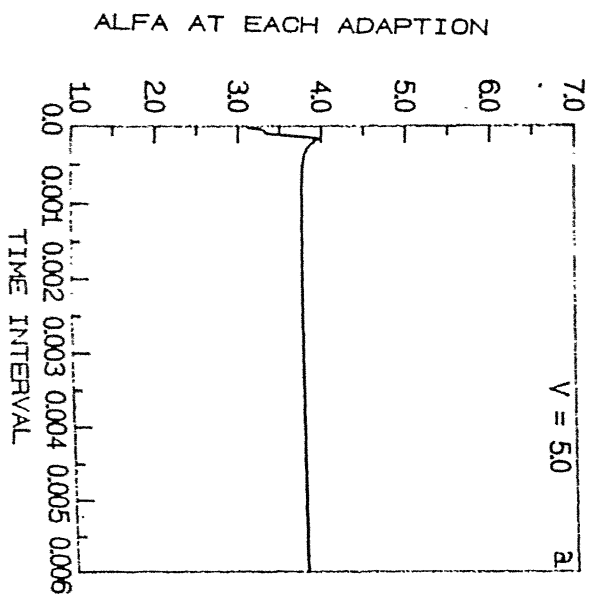
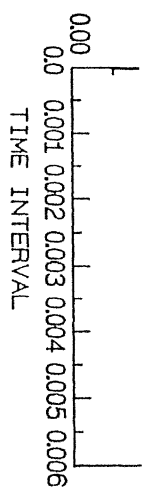
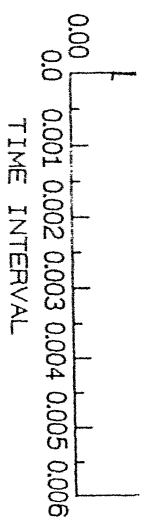
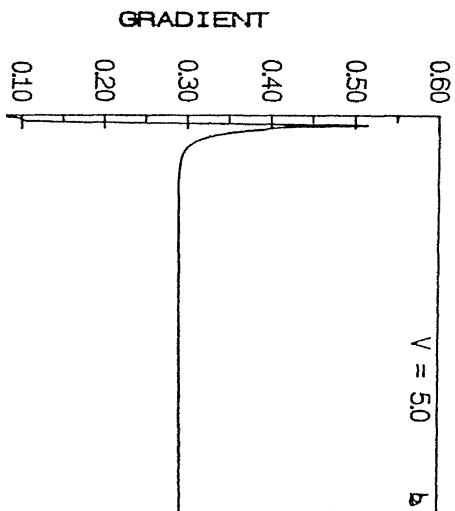
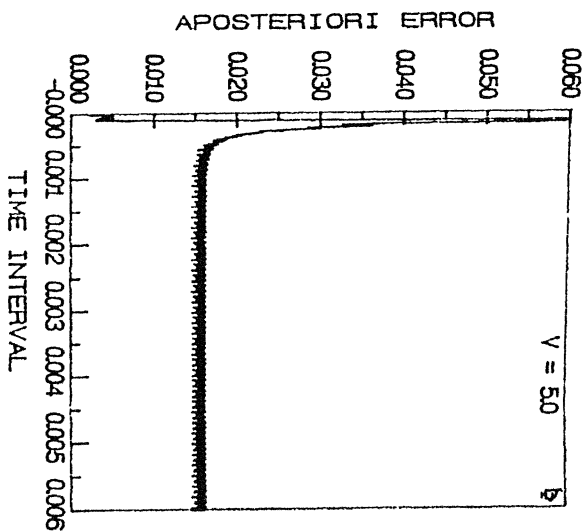
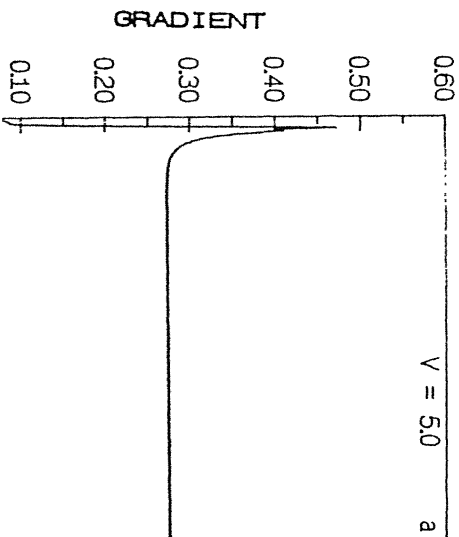
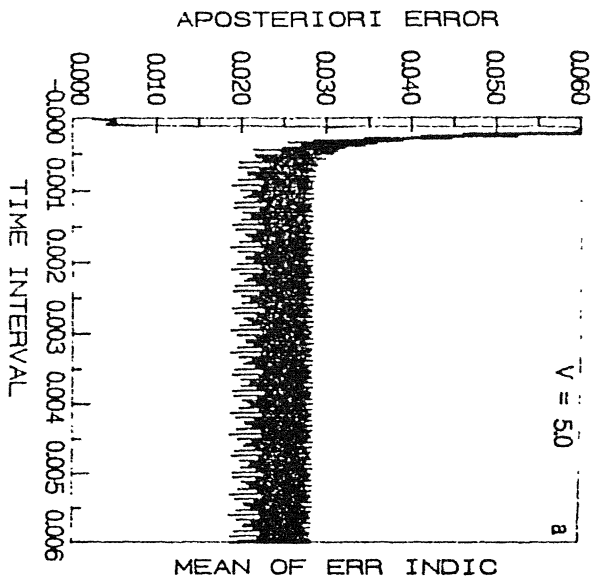


Fig. 527. FLAME PROBLEM : TIME EVOLUTION OF ERROR AND
ALFA VALUES (UNIFORM MESH (a) $N = 400$, (b) $N = 200$)



through the regular array of grid points in the uniform mesh. The predicted value of α for the uniform mesh (although it is not used here for grid adaption) increases during the ignition phase and attains a constant value during the steady propagation of the front. In fact the value of α also increases and decreases in accordance with the error behavior. Comparing the results of the $N = 200$ and $N = 400$ meshes, on the refined grid the amplitude of the oscillations in the *a posteriori* error is less and the value of α is also less. The smaller value of α can be attributed to the reduction in the *a posteriori* error as well as the asymptotic stabilization process in the estimation of the error.

The *a posteriori* error, α , solution gradient, and the maximum and mean of error indicators on adaptive meshes are plotted in Figs. 5.28-5.30 for different flow velocities. The value of α used at each time step decreases rapidly in the beginning and stabilizes for later times. Within each time step the magnitude of α for each adaption fluctuates slightly, due to the oscillations in the *a posteriori* error between the limits δ_1 and δ_2 . The oscillations in *a posteriori* error are caused by the following alternate sequence of events : (i) error increases when the flame front moves out of an existing optimal grid (ii) it decreases back due to the readjustment in the grid to the new location of the front after the adaption. The fluctuations in the *a posteriori* error are also reflected in other measures of error such as the maximum and the mean error indicators.

The norm of the solution gradient increases during the ignition phase and attains a perfectly constant value during the equilibrium phase of the front movement. In the test problem, the

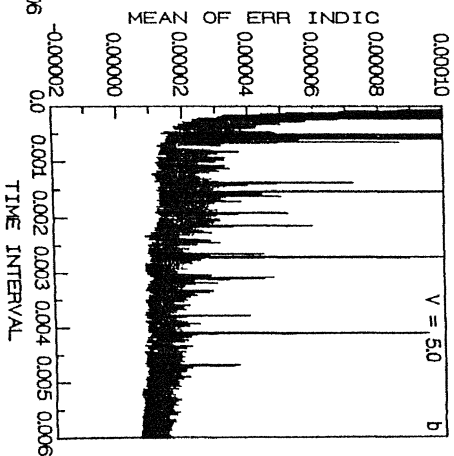
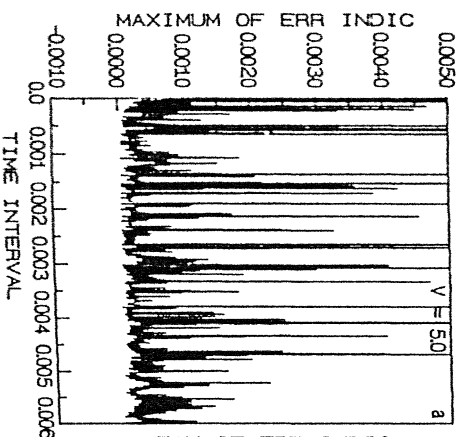
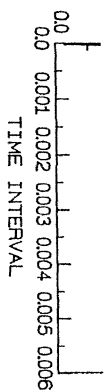
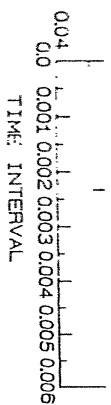
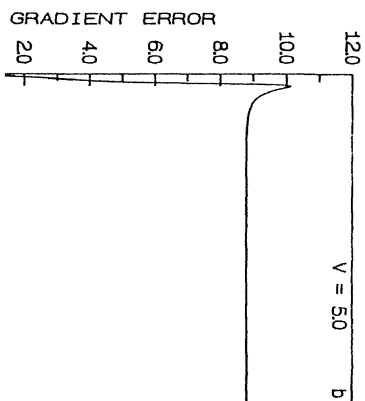
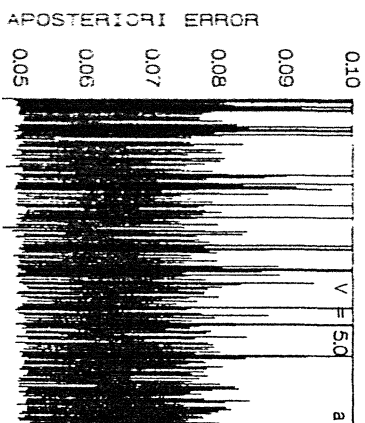
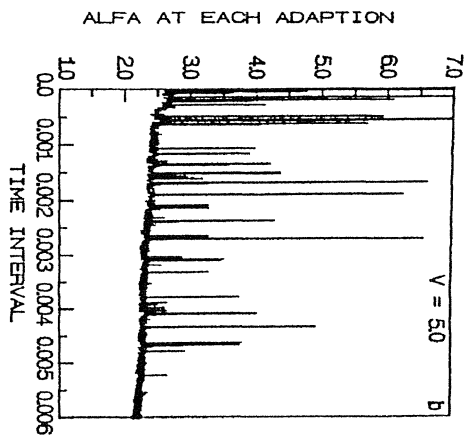
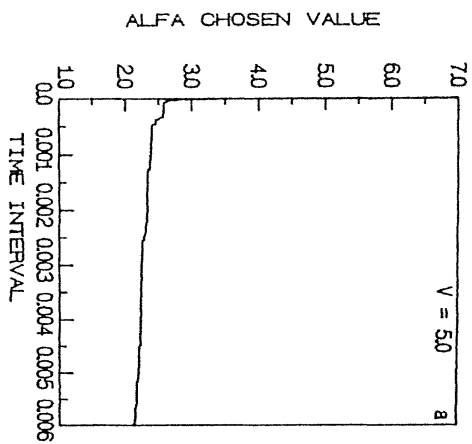


Fig. 5.28. FLAME PROBLEM : TIME EVOLUTION OF ALPHA VALUES,
GRADIENT AND ERRORS (h VERSION ; $V = 5.0$)



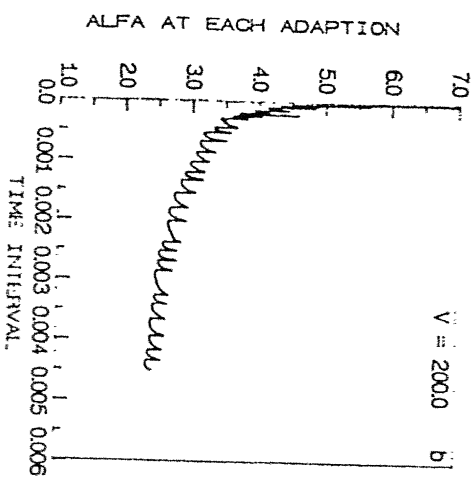
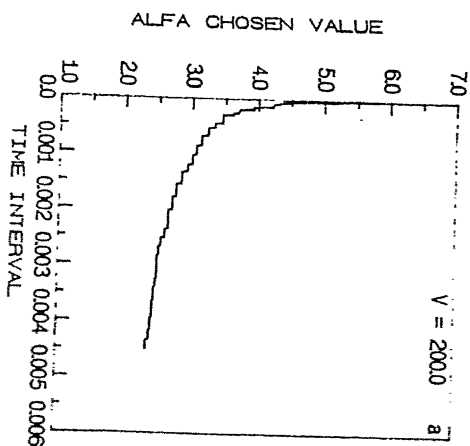
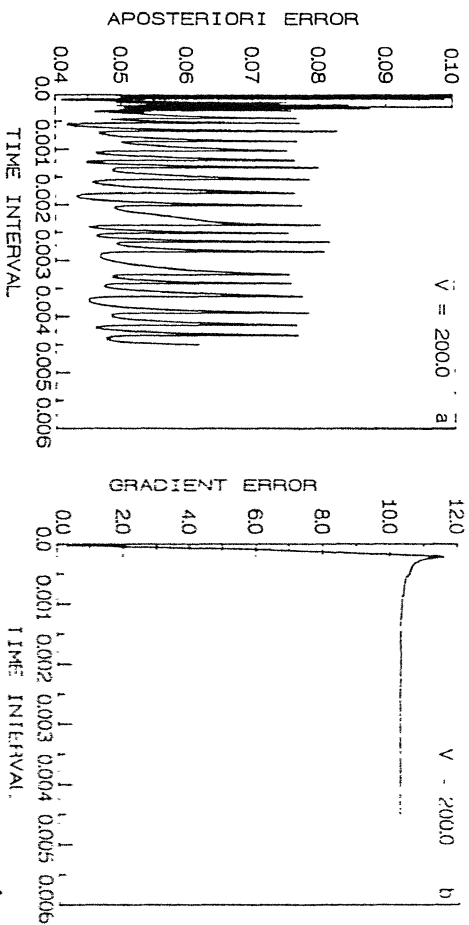
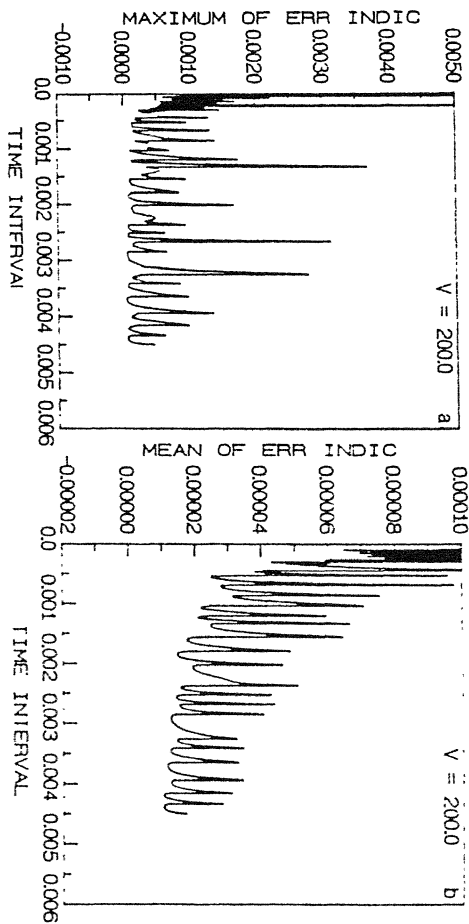
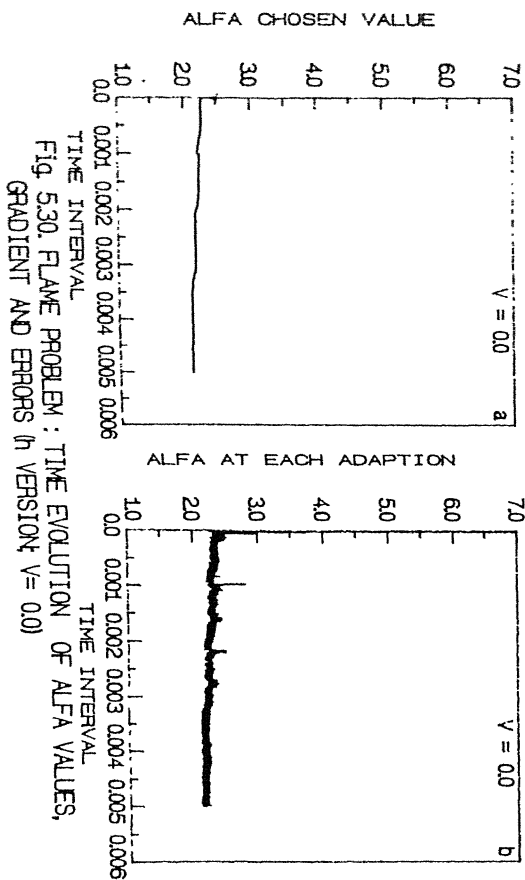
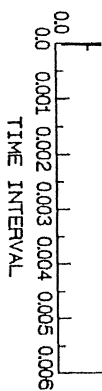
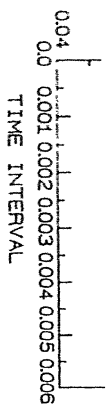
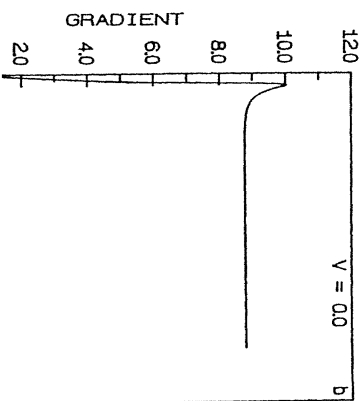
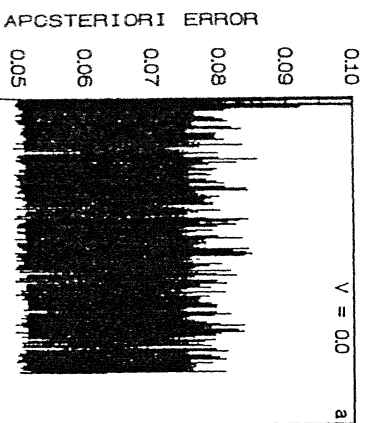
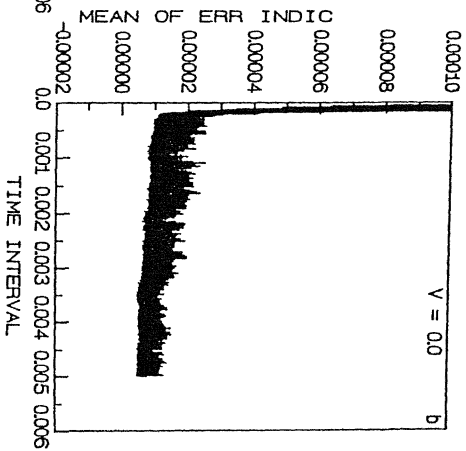
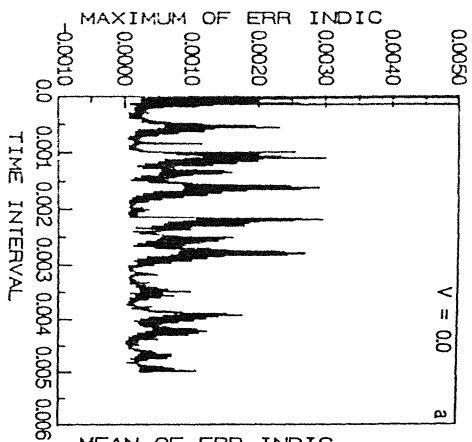


Fig 5.29. FLAME PROBLEM : TIME EVOLUTION OF ALFA VALUES,
GRADIENT AND ERRORS (h VERSION; $V = 200.0$)







norm of the gradient error was plotted because of the availability of the exact solution. For the flame propagation study, however, the norm of the gradient itself has been plotted since the norm of gradient error can not be estimated. The constancy of the gradient for later times is adequate proof for the efficiency of the α scheme and the suitability of the *a posteriori* error expression proposed in this Chapter.

The error behaviour is more or less same for all the flow velocities except for the following differences. For $V = 5.0$ situation, spikes are seen occasionally in the error plots and in the evolution of α . These spikes indicate temporary mismatch between the front location and the grid distribution due to the large time steps used in the calculation. However, the spikes are eliminated almost immediately in the next time step. Another effective means to obtain a smoother behaviour is to select a suitably small value of the time step Δt for the time marching. For the flow velocity of $V = 200.0$, the timewise oscillations of the error measures and the exponent α have much lower frequency. This implies that the flame movement is very slow in the presence of convection flow as discussed earlier.

In the Figs 5.31 - 5.36, the spatial distribution of error indicators are shown at various times for the situations of $V = 0.0$ and $V = 5.0$. The features discussed earlier in sections 5.6.2. are observed here also. For instance, sharp peaks in the error indicator correspond to the flame front location, while wide peaks represent inadequately refined elements outside the flame zone. The shifting of the sharp peak band with time indicates movement of flame front. For the flow velocity of $V = 5.0$, the

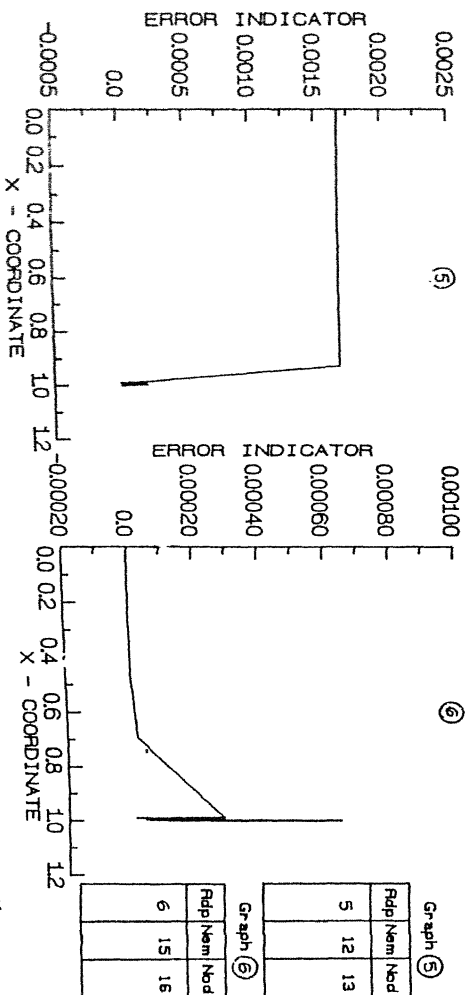
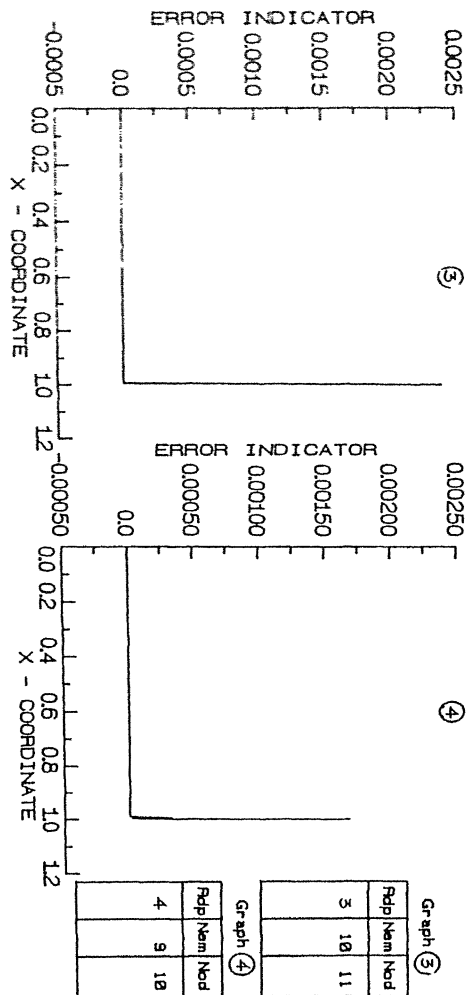
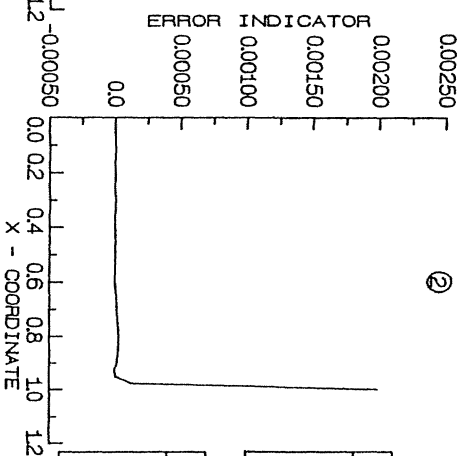
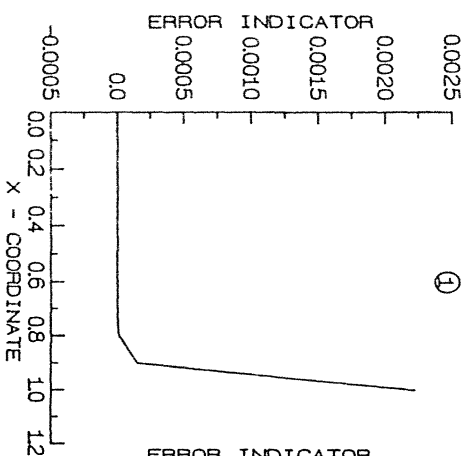


Fig. 5.31. FLAME PROBLEM : BEHAVIOUR OF ERRORS (h VERSION : $V = 0.0$)
($t = 0.000001$)



Graph ①

| Rdpt | Nem | Nod |
|------|-----|-----|
| 1 | 10 | 11 |
| | | |

Graph ②

| Rdpt | Nem | Nod |
|------|-----|-----|
| 2 | 9 | 10 |
| | | |

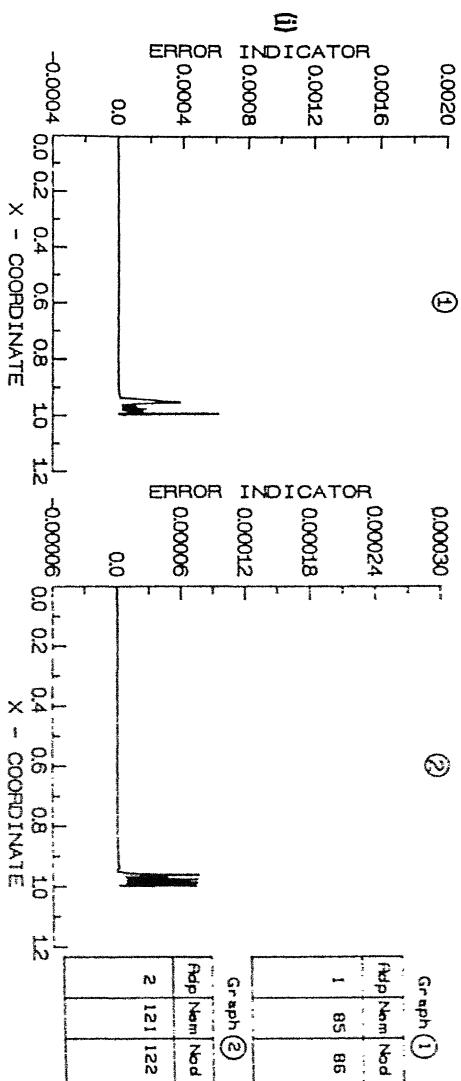
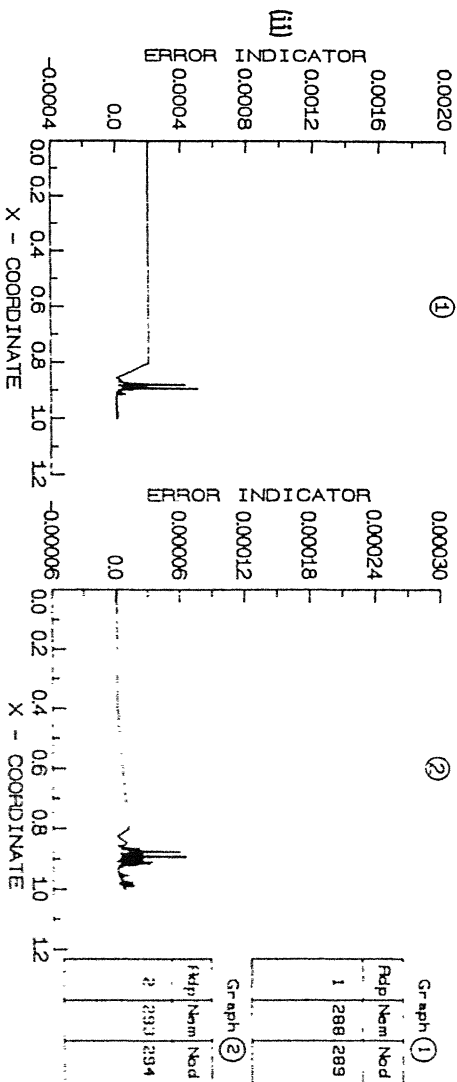
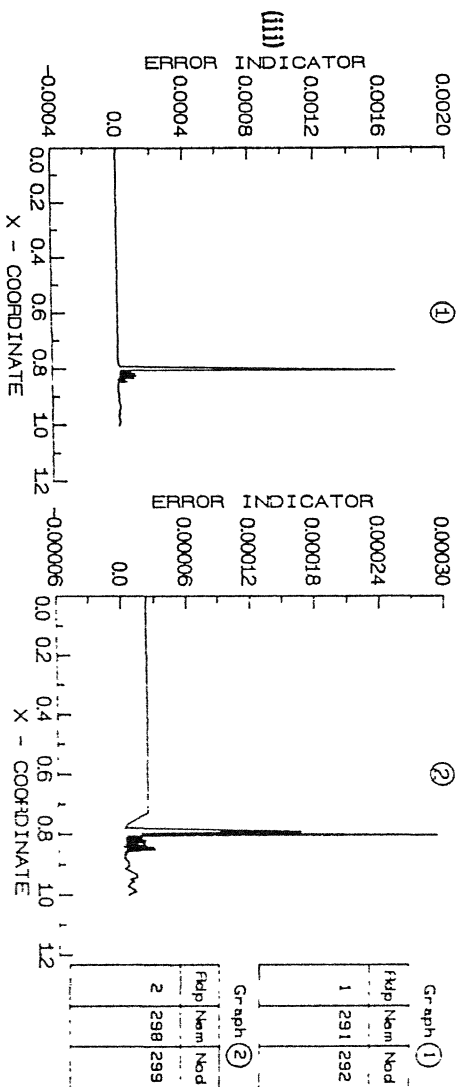


Fig. 5.32. FLAME PROBLEM : BEHAVIOR OF ERRORS IN VIBRATION : V (0.0)
 (i). $t = 0.00013$, (ii). $t = 0.00071$, (iii). $t = 0.00119$



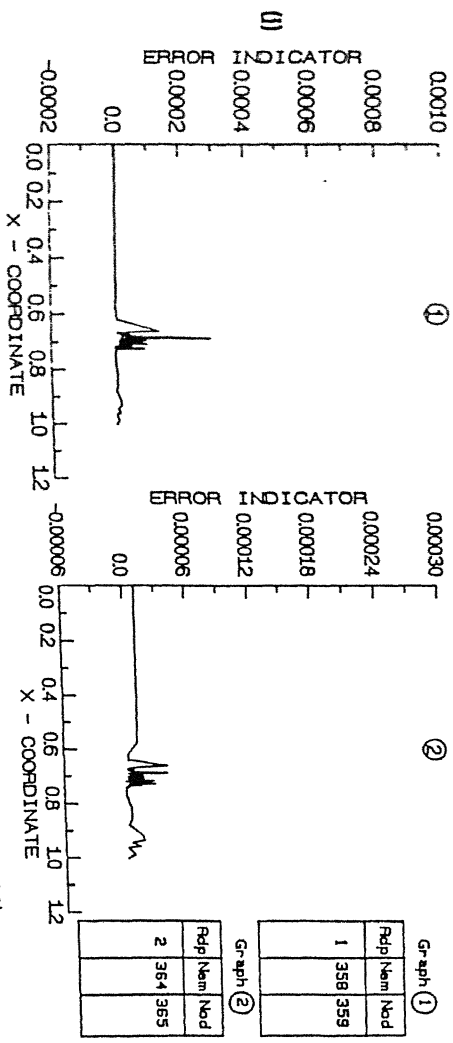
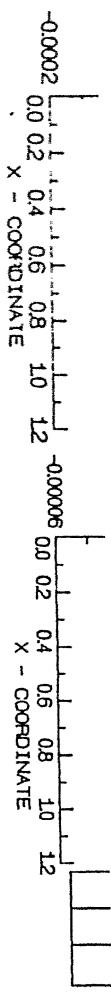
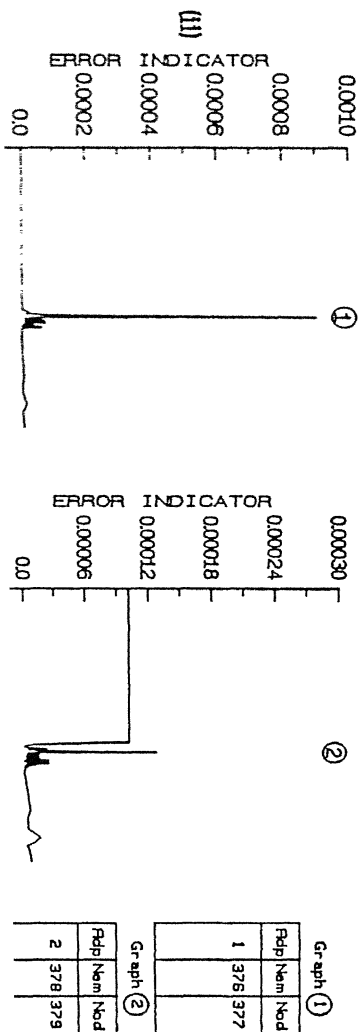
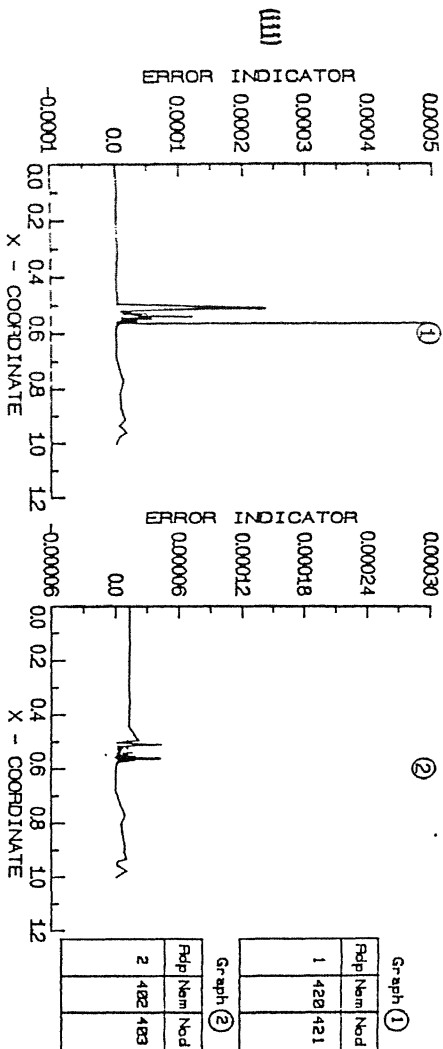
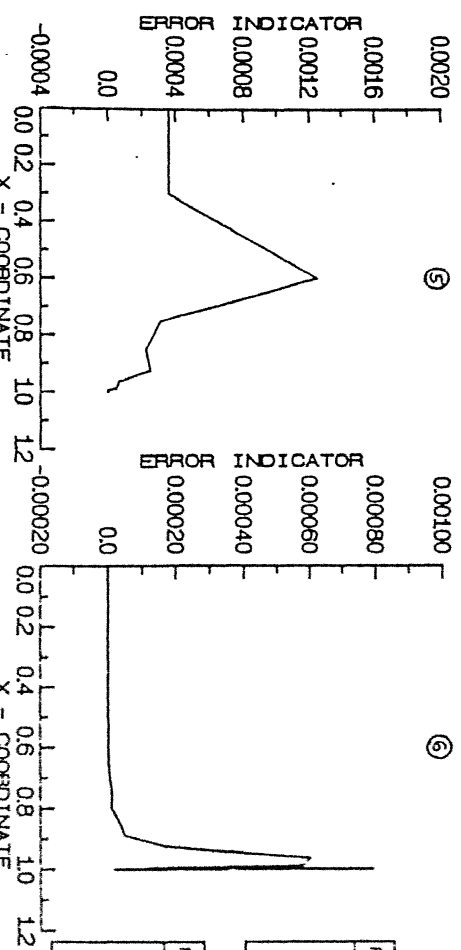
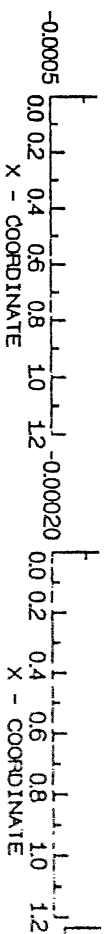


Fig. 5.33. FLAME PROBLEM : BEHAVIOUR OF ERRORS IN VERSION ; $V = 0.0$
 (i). $t = 0.00205$, (ii). $t = 0.00266$, (iii). $t = 0.00318$





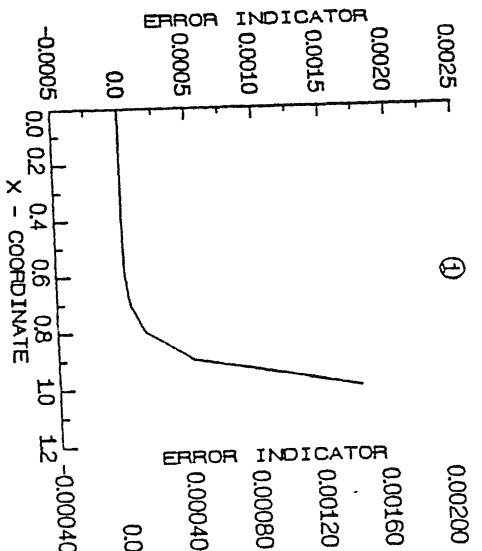
Graph ⑤

| Relp | Nem | Nbod |
|------|-----|------|
| 5 | 15 | 16 |

Graph ⑥

| Relp | Nem | Nbod |
|------|-----|------|
| 6 | 18 | 19 |

Fig. 5.34 FLAME PROBLEM : BEHAVIOUR OF ERRORS (n VERSION : v= 5.0)
(t = 0.00001)

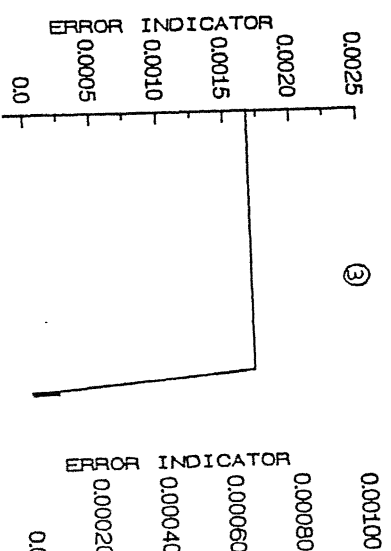
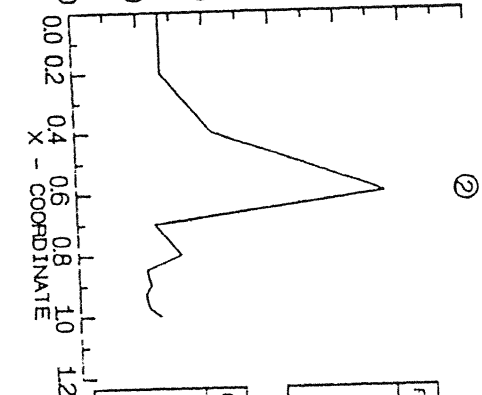


Graph ①

| Rdp Item | Nod |
|----------|-------|
| 1 | 10 11 |

Graph ②

| Rdp Item | Nod |
|----------|-------|
| 2 | 11 12 |

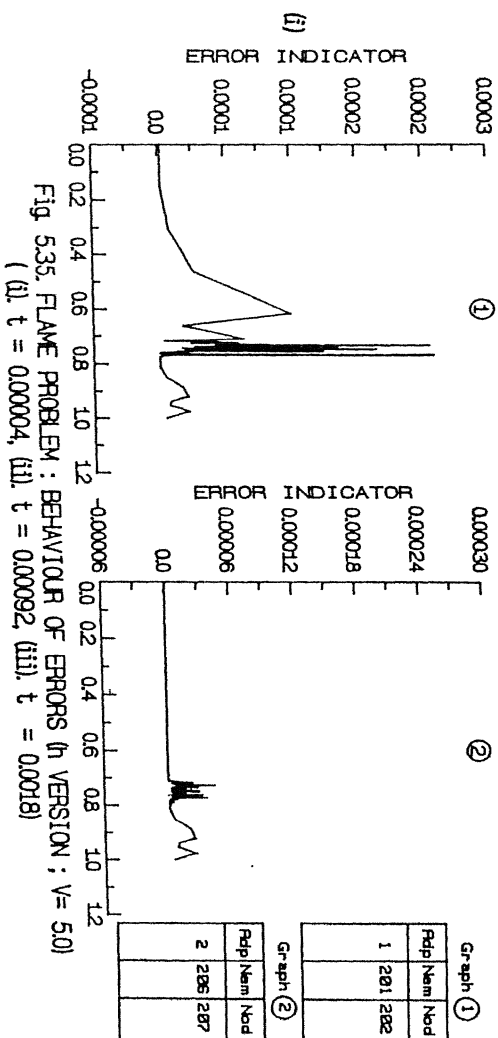


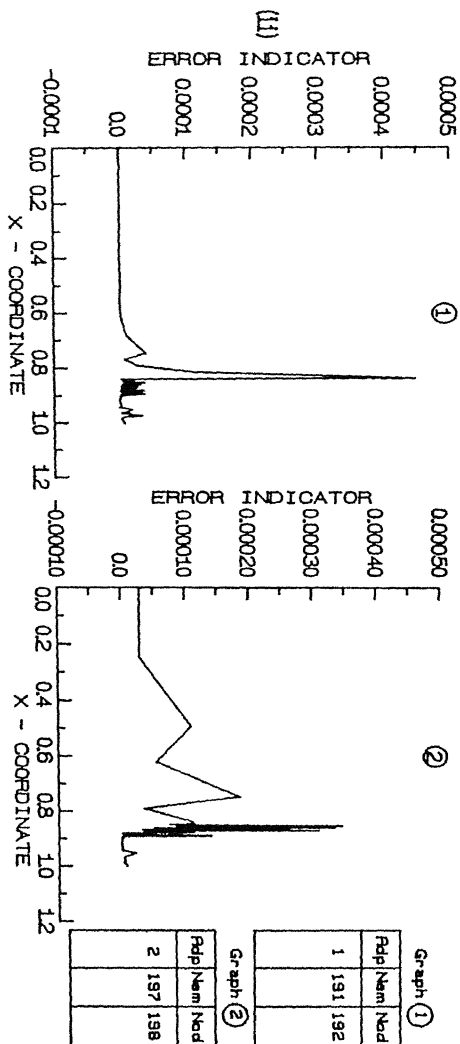
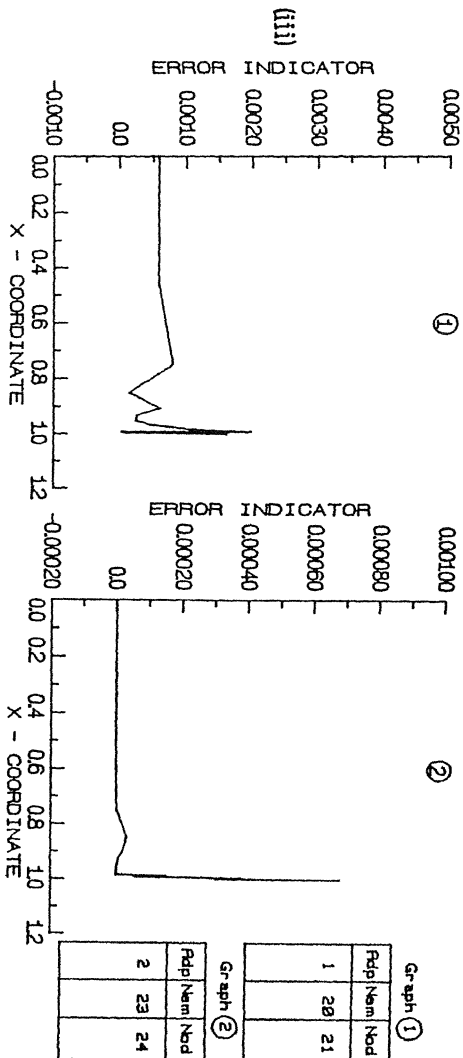
Graph ③

| Rdp Item | Nod |
|----------|-------|
| 3 | 12 13 |

Graph ④

| Rdp Item | Nod |
|----------|-------|
| 4 | 15 16 |





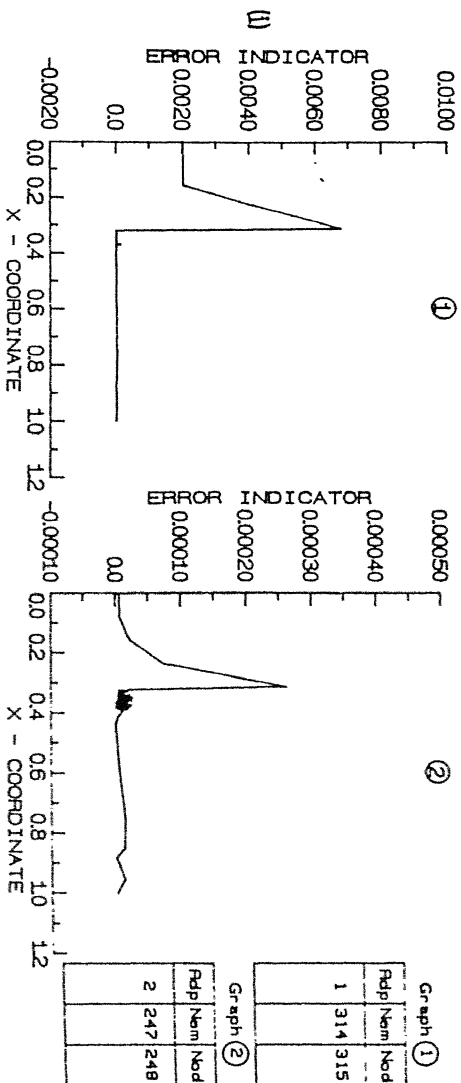
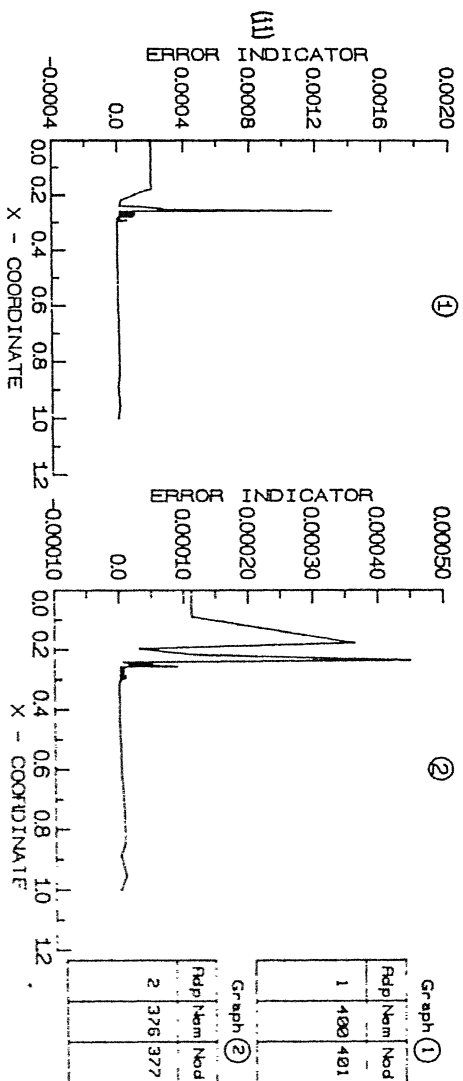
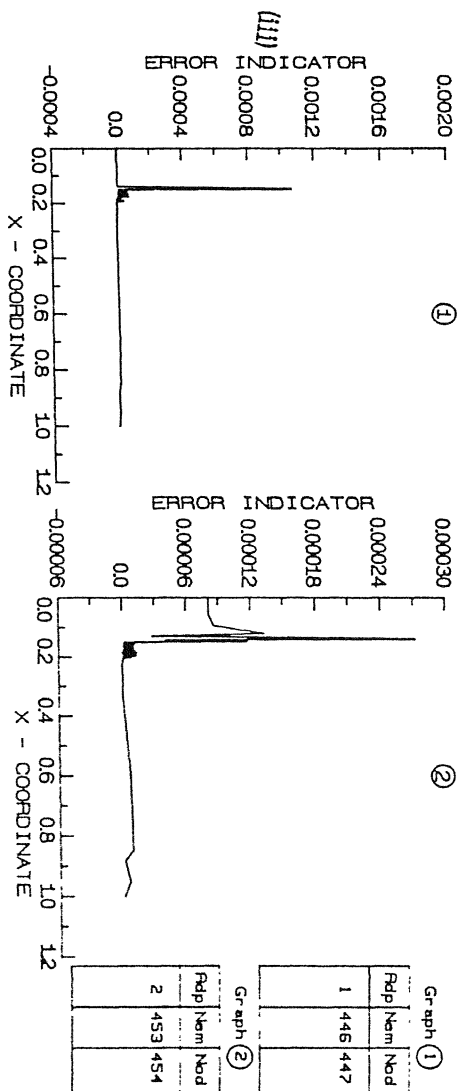


Fig 5.36. FLAME PROBLEM : BEHAVIOR OF ERRORS (n VERSION : $V = 5.0$)
 (i). $t = 0.00468$, (ii). $t = 0.00527$, (iii). $t = 0.0060$

wide peaks occur frequently at various time instants. This observation substantiates the earlier inference made in connection with Fig. 5.29 that perhaps a smaller time step may be necessary for eliminating the slight mismatch between the front location and the grid distribution.

5.7.3. Computational time :

The computational time for the h-version based α strategy has been presented in tabular form (table 5.3). In the algorithm, a special case $V = 0$ (in presence of flow) is considered and the efficiency of the adaptive method is investigated. Various quantities have been evaluated in a similar way as described in Chapter 4. Further, the source vector, the elementary matrices in the upwind formulation and the evaluation of α at each adaption for a given time may additionally contribute to the CPU time in the algorithm.

Initially the adaption is carried out starting with 10 uniform elements. Before ignition, the solution obtained by α strategy is very oscillatory because the mesh is coarse and larger time steps have been chosen. Consequently, the prediction of α is inaccurate and is a large value. Thus, the number of non-linear iterations performed is more in comparison to h version algorithm of Chapter 4. Due to large value of α , the adaption does not take place at every time step. As also, the less number of elements are required to satisfy the error criterion. Thus the CPU time requirements are less at time instants ($t = 0.00001$, $t = 0.00002$, 0.00004). It is observed that the bisection iterative method which has been employed here for the determination of α takes more time for the convergence when the number of elements is large.

| Dimension -less time | N | Nodes | Adp | Non-lin iter | CPU time/ time step (in seconds) |
|-------------------------|-----|-------|-----|-----------------|--|
| 0.000010 | 15 | 16 | 6 | 12 | 0.319785 |
| 0.000020 | 20 | 21 | 2 | 6 | 0.246980 |
| 0.000040 | 31 | 32 | 2 | 6 | 0.358685 |
| 0.000060 | 40 | 41 | 1 | 4 | 0.352592 |
| 0.000100 | 71 | 72 | 2 | 10 | 1.247347 |
| 0.000150 | 135 | 136 | 2 | 16 | 3.995194 |
| 0.000180 | 220 | 221 | 1 | 18 | 4.945962 |
| 0.000520 | 275 | 276 | 2 | 30 | 16.586933 |
| 0.001100 | 272 | 273 | 1 | 14 | 7.317469 |
| 0.001710 | 307 | 308 | 2 | 28 | 18.661804 |
| 0.003250 | 431 | 432 | 1 | 14 | 12.781802 |

Table. 5.3. CPU time of the h version based α strategy algorithm for the flame propagation ($V = 0.0$)

This is true near the ignition as well as in the flame front region. This increases CPU time for each non-adaption/adaption at each time step. Further, the dynamic setting of removal of tolerance in the refinement and derefinement processes may also affect the CPU time. Hence, at time instants ($t = 0.00006$, $t = 0.0001$, $t = 0.000150$, 0.000180 , $t = 0.000520$, 0.003250) the CPU time is very large although the number of elements is less. But in the present problem, the evaluation of upwind elementary matrices is not required at all. Thus, the h version based α strategy will show better performance if the calculation of upwinding matrices, selection of proper time step and the optimisation of the code are taken care off.

5.8. Conclusions :

The α -strategy described in the present Chapter has potential for being applied as a general purpose grid adaption scheme. It involves a self correcting error estimator whose asymptotic stabilization leads to the correct error estimate for the problem at hand. Results predicted by the α scheme match excellently with the analytical solution of the test problem having feature of front movement. For the non-linear flame problem also, the predictions of the α -strategy match with the corresponding h-version results for the no flow situation. It can be inferred from the behaviour of the various error measures that the error estimator of the α -strategy approaches the actual error in an asymptotic sense as h tend to zero in the front zone.

is developed for the use of a *posteriori* analysis. However, in order to validate the algorithm for adaptive refinement of grid, a problem with known analytical solution and possessing certain characteristic of the original problem, has been utilized as a test case.

An h-refinement algorithm for a system of partial differential equations in one space dimension, is developed. The method uses the concept of "eqidistribution" of errors for the residual as an adaptive criterion. A simple statistical procedure has been used for refinement of grids appropriate regions where the solution changes very rapidly. The mesh refinement strategy utilizes a *posteriori* bounds for a semi-discrete finite element set up.

In the hp refinement algorithm, h- and p-refinement algorithms have been conjointly applied for same classes of problems. Due to large number of grid adaptations and also the conversions from p- to h- version meshes, the hp method requires more additional calculations in comparison to h-refinement before ignition. However, after the ignition, h as well as hp methods perform well. Besides, the hp methods requires less number of elements to meet the specific tolerance criterion. The performance of h method is extremely good even at the initial time steps while in hp version, this may not be true as the approximation becomes increasingly sensitive to the quality of the initial mesh. Keeping processing cost, or storage limitations in mind, hp method performs well in comparison to h method only in the front region.

The algorithm, called dynamic grid strategy, has been applied to the flame problem in the presence of stationary mixture

and as well as to the test problem. It is an efficient method to solve for the grid and the solution. The advantage in this method lies in the fact that the grid distribution readjusts dynamically to concentrate points in regions of larger variation of the solution. The method performs better in case of front propagation problems in comparison to h and hp methods and is generally, computationally expensive.

(3). In addition to the above algorithms, a generalized h refinement based α strategy adaptive algorithm has been developed in the Chapter 5. The method has been applied to the flame propagation problem in presence of flow and for a test problem using upwind FE formulation. The method is based on equidistribution of error strategy involving local mesh size and the local residue of the FE solution. The method performs well for the transient problems, especially front propagation problems. This algorithm is applied to the flame problem in a stationary mixture and to a test problem considered in Chapter 4. A comparative study shows that this novel method performs reasonably well in comparison to other algorithms.

6.2. Suggestions for future work :

(1). The grid generation algorithm developed here can be applied to generate structured grid from the unstructured grid. A part of the algorithm developed here has been extended by Suzuki (1991) to generate structured surface grids based on unstructured grids.

(2). The method can be used to generate three dimensional

grids as well. Note that the grid spacing and adaptive grid generation can be achieved by taking appropriate control functions in the grid generation equations. The iterative method to obtain the grid by solving generation equations can be relaxed so that the ξ - and η -constant lines can be drawn only once.

(3). The h , h - p , and grid speed algorithms based on a posteriori error estimates can be used for the general reactive diffusive type problems in higher space dimensions. Especially, the refinement based on statistical method is very much useful in the front propagation problems as the method has a capabilities to generate dense meshes in the required zones. In higher dimensions, the combination of all adaptive strategies and the refinement based on the local error indicator expressions can be applied to variety of transient problems.

(4). For a general class of non-self adjoint linear and non-linear problems, the proposed self adjusting posteriori error estimate can be used in the adaptive refinement algorithms. The method is easily extendable to higher dimensions.

(5). The concept of dynamic and static zone analysis can also be introduced to the algorithm dealing with the h -version based α strategy. It may result in less expensive computation than the present work.

(6). For general class of one dimensional transient problems, the hp version based α strategy algorithm can also be developed in straight forward way.

References

- Acharya, S. and Patankar, S.V., 1985,
Use of an adaptive grid procedure for parabolic flow problems, *Int. J. of Heat and Mass Transfer*, Vol. 28, No. 6, pp. 1057-1066, (1985).
- Acharya, S. and Moukalled, F.H., 1990,
An adaptive grid solution procedure for convection-diffusion problems, *J. Comp. Phys.*, Vol. 91, pp. 32-54, (1990).
- Adjerid, S. and Flaherty, J. E., 1986a,
A moving mesh finite element method with local refinement for parabolic partial differential equations, *Comp. Meth. Appl. Mech. Engg.*, Vol. 55, No. 1-2, pp. 3-26, (1986).
- Adjerid, S. and Flaherty, J. E., 1986b,
A moving finite element method with error estimation and refinement for one dimensional time dependent partial differential equations, *SIAM. J. Numer. Analy.*, Vol. 23, No. 4, pp. 778-796, (1986).
- Adjedj, G. and Aubey, D., 1989,
Development of a hierarchical and adaptive finite element software, *Comp. Meth. Appl. Mech. Engg.*, Vol. 75, No. 1-3, pp. 153-166, (1989).
- Anderson, D. A. and Rai, M. M., 1982,
The use of solution adaptive grids solving partial differential equations, in J.F. Thompson (ed.), *Numerical Grid Generation*, North-Holland, pp. 317-338, (1982).
- Anderson, D. A., 1983a,
Adaptive mesh schemes based on grid speed, AIAA-83-1931, *AIAA 6th Computational Fluid Dynamic Conference*, Danvers, MA, (1983).
- Anderson, D. A., 1983b,
Adaptive grid methods for partial differential equations, in K.N. Ghia and U. Ghia (eds.), *Advances in Grid Generation*, FED - Vol. 5, ASME, pp. 1-15, (1983).
- Arcilla, A. S., 1991,
The third international conference in numerical grid generation in computational fluid dynamics and related fields, Univ. of Catalonia, Barcelona, Spain, June (1991).
- Arney, C. David and Flaherty, J. E., 1986,
A two dimensional mesh moving technique for time dependent partial differential equations, *J. Comp. Phys.*, Vol. 67, No. 1, pp. 124-144, (1986).
- Axelsson, O., 1976,
A class of iterative methods for finite element equations, *Comp. Meth. Appl. Mech. Engg.*, Vol. 9, pp. 123-137, (1976).

- Babuska, I. and Aziz, A. K., 1972,
Survey lectures on the mathematical foundations of the
finite element method, A. K. Aziz et al., (eds.), *The
Mathematical Foundations of the Finite Element Method in
Applications to Partial Differential Equations*, Academic
Press, New York, pp. 3-363, (1972).
- Babuska, I., 1975,
The self adaptive approach in the finite element method, in
J. R. Whiteman (ed.), *Mathematics of Finite Elements and
Applications*, Academic press, London, (1975).
- Babuska, I. and Rheinboldt, W. C., 1978a,
A posteriori error estimates for the finite element method,
Int. J. Num. Meth. Engg., Vol. 12, No. 10, pp. 1597-1615,
(1978).
- Babuska, I. and Rheinboldt, W. C., 1978b,
Error estimates for adaptive finite element computations,
SIAM. J. Numer. Analy., Vol. 15, No. 4, pp. 736-754, (1978).
- Babuska, I. and Rheinboldt, W. C., 1979a,
Analysis of optimal finite element meshes in R^1 , *Math. Comp.*,
Vol. 33, No. 146, pp. 435-463, (1979).
- Babuska, I. and Rheinboldt, W. C., 1979b,
Adaptive approaches and reliability estimations in finite
element analysis, *Comp. Meth. Appl. Mech. Engg.*, Vol. 17/18,
pp. 519-540, (1979).
- Babuska, I. and Rheinboldt, W. C., 1979c,
On the reliability and optimality of the finite element
method, *Comput. Struct.*, Vol. 10, No. 1, pp. 87-94, (1979).
- Babuska, I. and Rheinboldt, W. C., 1980,
Reliable error estimation and mesh adaption for the finite
element method, in J. T. Oden (ed.), *Computational Methods in
Nonlinear Mechanics*, North-Holland, Amsterdam, pp. 67-108,
(1980).
- Babuska, I. and Rheinboldt, W. C., 1981,
A posteriori error analysis of finite element solutions of
one dimensional problems, *SIAM. J. Numer. Analy.*, Vol. 18,
No. 3, pp. 565-589, (1981).
- Babuska, I. and Dorr, M. R., 1981,
Error estimates for the combined h and p versions of the
finite element method, *Numer. Math.*, Vol. 37, pp. 257-277,
(1981).
- Babuska, I., Szabo, B. A. and Katz, I.N., 1981,
The p-version of the finite element method, *SIAM J. Numer.
Analy.*, Vol. 18, No. 3, pp. 515-545, (1981).
- Babuska, I. and Rheinboldt, W. C., 1982,
A survey of a posteriori error estimators and adaptive

approaches in the finite element method, *Technical Note BN - 981, Laboratory for Numerical Analysis, Univ. of Maryland*, (1982).

Babuska, I and Szymczak, W. G., 1982,
An error analysis for the finite element method applied to convection diffusion problems, *Comp. Meth. Appl. Mech. Engg.*, Vol. 31, pp. 19-42, (1982).

Babuska, I., Chandra, J. and Flaherty, J. E., 1983,
Adaptive computational methods for partial differential equations, SIAM, Philadelphia, (1983).

Babuska, I., Gui, W. and Szabo, B., 1984,
Performance of the h, p and h-p versions of the finite element method, technical note BN-1027, *Laboratory for Numerical Analysis, Univ. of Maryland, Maryland*, (1984).

Babuska, I., 1986,
Feedback adaption and a posteriori estimates in I. Babuska, O. C. Zienkiewicz, and adaptive refinements in finite element computations, J. Gago and E. R. De A. Oliveria (eds.), *Accuracy Estimates and Adaptive Refinements in Finite Element Computations*, John Wiley and Sons, New York, pp. 3-24, (1986).

Babuska, I. and Gui, W., 1986,
Basic principles of feedback and adaptive approaches in the finite element method, *Comp. Meth. Appl. Mech. Engg.*, Vol. 55, No. 1-2, pp. 27-42, (1986).

Babuska, I. and Luskin, M., 1989,
An adaptive time discretization procedure for parabolic problems, in R. Vichnevetsky and R. S. Stepleman, (eds.), *Proc. of IMACS Internat. Symp.*, IMACS, pp. 5-9, (1989).

Bachmann, P.L., Wittchen, S.L., Shepard, M.S., Grice, K.R. and Yerry, M.A., 1987
Robust geometrically based, automatic two-dimensional mesh generation, *Int. J. Num. Meth. Engg.*, Vol. 24, pp. 1043-1078, (1987)

Bank, R. E. and Sherman, A. H., 1980,
A refinement algorithm and dynamic data structure for finite element meshes, Report CNATR-166, Univ. of Texas at Austin, (1980).

Bank, R. E., 1983,
The efficient implementation of local mesh refinement algorithms, in I. Babuska, J. Chandra and J. E. Flaherty., (eds.), *Adaptive Computational Methods for Partial Differential Equations*, SIAM, Philadelphia, pp. 74-81, (1983).

Bank, R.E. and Weiser, A., 1985,
Some a posteriori error estimates for elliptic partial differential equations, *Math. Comp.*, Vol. 44, No. 170, pp.

283-301, (1985).

- Bank, R. E., 1986,
Analysis of a local a posteriori error estimate for elliptic equations, in I. Babuska, O. C. Zienkiewicz, J. Gago and E. R. De A. Oliveria (eds.), *Accuracy Estimates and Adaptive Refinements in Finite Element Computations*, John Wiley and Sons, New York, pp. 119-128, (1986).
- Barrett, J.W. and Morton, K.W., 1980,
Optimal finite element solutions to diffusion convection problems in one dimension, *Int. J. Num. Meth. Engg.*, Vol. 15, (1980).
- Benkhaldoun, F. and Larrousturou, B., 1987,
Explicit adaptive calculations of wrinkled flame propagation, *Int. J. Num. Meth. Engg.*, Vol. 7, pp. 1147-1158, (1987).
- Benkhaldoun, F., Dervieux, A., Fernandez, G., Guillard, H. and Larrousturou, B., 1988a,
Some finite element investigations of stiff combustion problems : Mesh adaption and implicit time stepping, in CM. Brawner and C. Schmidt-Laine, (eds.), *Mathematical modelling in Combustion and Related Topics*, NATO AST Series : Applied Science - Vol. 140, Martinus Nijhoff Publishers, Dordrecht, pp. 339-410, (1988).
- Benkhaldoun, F., Leyland, and Larrousturou, B., 1988b,
Dynamic mesh adaption for unsteady nonlinear phenomena - application to flame propagation, in Sengupta, et al. (eds.), *Numerical Grid Generation in Computational Fluid Mechanics* 88, Pineridge Press, Swansea, UK, pp. 63-74, (1988).
- Benkhaldoun, F. and Larrousturou, B., 1989,
Finite element adaptive investigation of curved stable and unstable flame front, *Comp. Meth. Appl. Mech. Engg.*, Vol. 76, No. 2, pp. 119-134, (1989).
- Berger, M.J., 1987
Adaptive finite difference methods in fluid dynamics, UC-32, No. DOE/ER/03077-277, Research and development Report, Courant Mathematics and Computing Laboratory, New York University, February, (1987)
- Bermudez. J., Durany, P. M. and Vazquez. C., 1989,
An upwind model for solving transport diffusion reaction systems, *Int. J. Num. Meth. Engg.*, Vol. 28, pp. 2021-2039, (1989).
- Bieterman, M. and Babuska, I., 1982a,
The finite element method for partial differential equations, I.A - posteriori error estimation, *Numer. Math.*, Vol. 40, pp. 339-371, (1982).
- Bieterman, M. and Babuska, I., 1982b,
The finite element method for partial equations, II. the adaptive mesh selection, *Numer. Math.*, Vol. 40, pp. 372-406,

(1982).

- Bieterman, M. and Babuska, I., 1986,
An adaptive method of lines with error control for
parabolic equations of reaction-diffusion type, *J. Comp.
Phys.*, Vol. 63, No. 1, pp. 33-66, (1986).
- Bjer, C. I., 1989,
Adaptive mesh in dynamic problems by space time approach,
Comp. and Struct., Vol. 33, No. 2, pp. 319-326, (1989).
- Blacker, T. D., and Stephenson, M. B., 1991,
Paving : A new approach to automated quadrilateral mesh
generation : *Int. J. Meth. Engg.*, Vol. 32, No 4, pp. 811-848,
(1991).
- Brackbill, 1982,
Coordinates system control: adaptive meshes, in J. F.
Thompson (ed.), *Numerical Grid Generation*, North Holland,
Amsterdam, pp. 277-294, (1982).
- Brawner, CM and Schmidt-Laine, C., 1988,
Mathematical modelling in combustion and related topics, NATO
AST Series : Applied Science - Vol. 140, Martinus Nijhoff
Publishers, Dordrecht, (1988).
- Brooks, A. N. and T.J.R Hughes, 1982,
Streamline upwind/Petrov-Galerkin formulations for convection
dominated flows with particular emphasis on the
incompressible Navier-Stokes equation, *Comp. Methods Appl.
Mech. Engg.*, Vol. 32, pp. 199-259, (1982).
- Buckmaster, J. D. and Ludford, G. S. S., 1982,
Theory of laminar flames, Cambridge Univ. Press, Cambridge,
(1982).
- Carey, G. F., 1977,
An analysis of finite element equations and mesh
subdivision, *Comp. Meth. Appl. Mech. Engg.*, Vol. 11, pp.
93-105, (1977).
- Carey., G. F., 1979,
Adaptive refinement and nonlinear fluid problems, *Comp. Meth.
Appl. Mech. Engg.*, Vol. 17/18, pp. 541-560, (1979).
- Carey, G. F. and Humphrey, D., 1981,
Mesh refinement and iterative solution methods for finite
element computations, *Int. J. Num. Meth. Engg.*, Vol. 17, No.
11, pp. 1717-1734, (1981).
- Carey, G. F. and Plover, T., 1983,
Variable upwinding and adaptive mesh refinements
in convection-diffusion, *Int. J. Num. Meth. Engg.*, Vol. 19,
No. 3, pp. 341-353, (1983).

- Carey, G.F. and Oden, J. T., 1984a,
Finite Elements : Computational Aspects,
 Vol. 3, Prentice Hall, New Jersey, (1984).
- Carey, G.F. and Oden, J. T., 1984b,
Finite Elements : Fluid Mechanics,
 Vol. 6, Prentice Hall, New Jersey, (1984)
- Carey, G. F., 1988,
 Some aspects of adaptive grid computations, *Comput. Struct.*,
 Vol. 30, No. 1/2, pp. 297-302, (1988).
- Cavendish, J.C., Field, D.A. and Frey, W.H., 1984
 An approach to automatic three dimensional finite element
 mesh generation, *Int. J. Num. Meth. Engg.*, Vol. 21, pp.
 329-347, (1985)
- Chargy, D., Dervieux, A. and Larrousturou, B., 1990,
 Upwind adaptive finite element investigations of the
 two-dimensional reactive interaction of supersonic gaseous
 jets, *Int. J. Num. Meth. Engg.*, Vol. 11, No. 6, pp. 751-756,
 (1990).
- Christie, J. and Mitchell, A.R., 1978,
 Upwinding of high order Galerkin methods in conduction
 convection problems, *Int. J. Num. Meth. Engg.*, Vol. 12, pp.
 1764-1771, (1978).
- Ciarlet, P. G., 1978,
The finite element method for elliptic problems,
 North-Holland Publ. Co., Amsterdam, (1978).
- Coyle, J. M., Flaherty, J. E., and R. Ludwig, 1986
 On the stability of mesh equidistribution strategies
 for time-dependent partial differential equations, *J. Comp.*
Phys., Vol. 62, No. 1, pp. 26-39, (1986).
- Davis, S. F. and Flaherty, J. E., 1982,
 An adaptive finite element method for initial-boundary value
 problems for partial differential equations, *SIAM J. Sci.*
Stat. Comput., Vol. 3, No. 1, pp. 6-27, (1982).
- Demkowicz, L. and Oden, J. T., 1986,
 An adaptive characteristic Petrov-Galerkin finite element
 method for convection dominated linear and nonlinear
 parabolic problems in two space variables, *Comp. Meth. Appl.*
Mech. Engg. Vol. 55, pp. 67-89, (1986).
- Demkowicz, L., Oden, J. T., Rachowicz, W. and Hardy, 1989,
 Toward a universal h-p adaptive finite element strategy, part
 1 : constrained approximation and data structure, *Comp. Meth.*
Appl. Mech. and Engg., Vol. 77, No. 1-2, pp. 79-112, (1989).
- Dervieux, A., Larrousturou, B. and Peyret, R., 1989,
 On some adaptive numerical approaches of thin flame
 propagation problems, *Computers and Fluids*, Vol. 17, No. 1,
 pp. 39-60, (1989).

- Dervieux, A. and Larrouturnou, B., 1990,
Numerical combustion, lecture notes in physics, Vol. 351,
 Springer Verlag, (1990).
- Do Carmo, E. G. D. and Galeao, A. C., 1991,
 Feedback Petrov-Galerkin method for convection dominated
 problem, *Comp. Meth. Appl. Mech. Engg.*, Vol. 88, No.1, pp.
 1-16, (1991).
- Douglas, J. and Russell, T., (1982),
 Numerical methods for convection dominated diffusion problems
 based on combining the method of characteristics with finite
 element or finite difference procedures, *SIAM J. Num. Anal.*,
 Vol. 19, pp. 871-885, (1982).
- Dupont, T., 1982,
 Mesh modification for evolution equations, *Math. Comput.*,
 Vol. 39, No. 159, pp. 85-107, (1982).
- Dwyer, H. A., Kee, R. J. and Sanders, B. R., 1980,
 Adaptive grid method for problems in fluid mechanics and
 heat transfer, *AIAA. J.*, Vol. 10, pp. 1205-1212, (1980).
- Dwyer, H. A., Smooke, M. D. and Kee, R. J., 1982,
 Adaptive gridding for finite difference solution to heat and
 mass transfer problems, in J. F. Thompson (ed), *Numerical
 Grid Generation*, North Holland, Amsterdam, pp. 339-356,
 (1982).
- Dwyer, H. A. and Sanders, B. R., 1983,
 Ignition and flame propagation studies with adaptive
 numerical grids, *Combustion and Flames*, Vol. 52, No. 1, pp.
 11-23, (1983).
- Dwyer, H. A., 1983,
 A discussion of some criteria for the use of adaptive
 gridding, in I. Babuska, J. Chandra and J. E. Flaherty.,
 (eds.), *Adaptive computational methods for partial
 differential equations*, SIAM, Philadelphia, pp. 111-122,
 (1983).
- Eiseman, P. R., 1983,
 Adaptive grid generation by mean value relaxation, in K. N.
 Ghia and U. Ghia (eds.), *Advances in Grid Generation*,
 ASME-FED, Vol. 5, pp. 29-34, (1983)
- Eiseman, P. R. 1985a,
 Grid generation for fluid mechanics computations, *Annual
 Review of fluid mechanics*, Vol. 17, pp. 487-522, (1985).
- Eiseman, P. R., 1985b,
 Solution adaptivity using a triangular mesh : The
 free-Lagrange method, *lecture notes in physics*, Vol. 238,
 Springer-Verlag, New York, pp. 206-235, (1985).

- Eiseman, P. R., 1986,
Adaptive grid generation, *Comp. Meth. Appl. Mech. Engg.*, Vol. 64, Nos. 1-3, pp. 301-320, (1986).
- Eiseman, P. R. and Eriebacher, G., 1987,
Grid generation for the solution of partial differential equations, *NASA contractor report 178365*, ICASE report no. 87-57, Langley research Center, Virginia, (1987).
- Eriebacher, G. and Eiseman, P. R., 1987,
Adaptive triangular mesh generation, *AIAA. J.*, Vol. 25, No. 10, pp. 1356-1364, (1987).
- Eriksson, K., 1985,
Improved accuracy by adapted mesh-refinement in the finite element method, *Math. Comp.*, Vol. 44, No. 170, pp. 321-343, (1985).
- Eriksson, K. and Johnson, C., 1987,
Error estimates and automatic time step control for non-linear parabolic problems, *SIAM. J. Numer. Anal.*, Vol. 24, No. 1, pp. 12-23, (1987).
- Eriksson, K. and Johnson, C., 1988,
An adaptive finite element method for linear elliptic problems, *Math. Comp.*, Vol. 50, No. 182, pp. 361-383, (1988).
- Eriksson, K. and Johnson, C., 1990,
Adaptive streamline diffusion finite element methods for convection-diffusion problems, Rep. No. 1990-18/ISSN 0347-2809, Department of Mathematics, Chalmers University of Technology and the University of Goteborg, Sweden, (1990).
- Eriksson, K. and Johnson, C., 1991,
Adaptive finite element methods for parabolic problems 1 : A linear model problem, *SIAM. J. Numer. Anal.*, Vol. 28, No. 1, pp. 43-77, (1991).
- Flaherty, J.E., Coyle, J.J., Ludwig, R. and Davis, S. F., 1983,
Adaptive finite-element methods for parabolic partial differential equations, in I. Babuska, J. Chandra and J. E. flaherty (eds.), *Adaptive Computational Methods for partial differential equations*, SIAM, Philadelphia, pp. 1621-1656, (1983).
- Gago, J. P., Zienkiewicz, Kelly, D. W. de S. R. and Babuska, I., 1983,
Aposteriori error analysis and adaptive processes in the finite element method : Part II - adaptive mesh refinement, *Int. J. Num. Meth. Engg.*, Vol. 19, No. 11, pp. 1621-1656, (1983).
- Gelinas, R. J., Doss, S. K. and Miller, K., 1981,
The moving finite element method applications to general partial differential equations with multiple large gradients, *J. Comp. Phys.*, Vol. 40, No. 1, pp. 202-249, (1981).

George, P. L., Hecht, F. and Saltel, E., 1992,
Automatic mesh generator with specified boundary, *Comp. Meth.
Appl. Mech. Engg.*, Vol. 92, No. 3, pp. 269-288, (1992).

Ghia, K., Ghia, U., 1983,
Advances in Grid Generation, ASME-FED, Vol. 5, (1983).

Ghia, K., Ghia, U. and Shin, C. T., 1983
Adaptive grid generation for flows with local high gradient
regions, in K. N. Ghia and U. Ghia, *Advances in Grid
Generation*, ASME-FED, Vol. 5, pp. 29-34, (1983).

Giovangigli, V. and Smooke, M. D., 1989,
Adaptive continuation algorithms with application to
combustion problems, *Appl. Numer. Math.*, Vol. 5, pp. 305-331,
(1989).

Glowinski, R., 1985,
Numerical Simulation of Combustion Phenomena., R. Glowinski,
Larroutturou, B. and Teman R., eds., lecture
notes in Physics 241, Springer-Verlag, NY, (1985)

Gnoffo, P. A., 1982,
A vectorized finite volume, adaptive grid algorithm applied
to planetary entry problems, *AIAA- 82 1018*, *AIAA/ASME 3rd
Joint Thermo physics, Fluids, Plasma and Heat Transfer
Conference*, St. Louis, MO, (1982).

Greenberg, J. B., 1985,
A new self adaptive grid method, *AIAA. J.*, Vol. 23, No. 2,
pp. 317-320, (1985).

Gui, W. and Babuska, I., 1986a,
The h, p and h-p Versions of the finite element method in 1
dimension, part I. The error analysis of the p-Version ,
Numer. Math, Vol. 49, pp. 577-612, (1986)

Gui, W. and Babuska, I., 1986b,
The h, p and h-p Versions of the finite element method in 1
dimension, part II. The error analysis of the h and h-p
Version, *Numer. Math*, Vol. 49, pp. 613-657, (1986)

Gui, W. and Babuska, I., 1986c,
The h, p and h-p Versions of the finite element method in 1
dimension, part III. The Adaptive h-p Version ,
Numer. Math, Vol. 49, pp. 659-683, (1986)

Hawken, D. F, 1987,
Review of adaptive-grid techniques for solution of
partial differential equations, *Progress in Aerospace
Research*, Vol. 24, pp. 29-49, (1987).

Ho-Le, K., 1988,

- Finite element mesh generation methods : A review and classification, *Comp. Aided Des.*, Vol. 20, pp. 27-38, (1988).
- Hughes, T. J. R., 1978,
A simple scheme for developing upwind finite elements, *Int. J. Num. Meth. Engg.*, Vol. 12, No. 9, pp. 1359-1365, (1978).
- Irons, B.M., 1970,
A frontal solution progrm for finite elements, *Int. J. Numer. Meth. Engg.*, Vol. 2, pp. 5-32, (1970).
- Joe, B. and Simpson, R. B., 1986,
Triangular meshes for regions of complicated shape, *Int. J. Num. Meth. Engg.*, Vol. 23, No. 5, pp. 751-778, (1986).
- Johnson, C., 1989,
Adaptive finite element methods for diffusion and convection problems, *Computer Methods in Appl. Mech. and Engg.*, Proc. of workshop on Reliability in Computational Mechanics, Austin, November, (1989).
- Kapila, A. K., 1980,
Reactive-diffusive systems with Arrhenius kinetics : dynamics of ignition, *SIAM. J. Numer. Math.*, Vol. 39, pp. 21-36, (1980).
- Kela, A., Perruchio, R. and Voelker, 1986
Toward automatic finite element analysis, *Comp. Mech. Eng*, Vol. 5, No. 1, (1986)
- Kelly, D. W., de S. R. Gago, J. P., Zienkiewicz, O. C. and Babuska, I., 1983,
Aposteriori error analysis and adaptive processes in the finite element method : Part I - Error Analysis, *Int. J. Num. Meth. Engg.*, Vol. 19, No. 11, pp. 1593-1619, (1983).
- Kreis, R. I., Thames, F. C, and Hasson, H. A., 1986,
Application of a variational method for generating adaptive grids, *AIAA. J.*, Vol. 24, No. 3, pp. 404-410, (1986).
- Larrouturou, B., 1985,
Numerical simulation of flame propagation , in R. Glowinski, B. Larrouturou and R. Temam (eds.), *Numerical Simulation of Combustion Phenomena*, Numerical Simulation of Combustion Phenomena., Lecture notes in Physics, Vol. 241, Springer - Verlag, pp. 300-312, (1985).
- Larrouturou, B., 1986,
Adaptive numerical methods for unsteady flame propagation, G. S. S. Ludford (ed.), *Combustion and Chemical Reactors*, Lectures in Applied Mathematics, Vol. 24, No. 2, pp. 415-435, (1986).
- Lee, D.N. and Ramos, J.I., 1983,

- Application of the finite-element method to one-dimensional flame propagation problems, *AIAA. J.*, Vol. 21, pp. 262-269, (1983).
- Lee, S. H. and Hsieh, S. S., 1991,
Applications of a self adaptive algorithm to non-linear finite element analysis, *Int. J. Num. Meth. Engg.*, Vol. 32, No. 5, pp. 105-1077, (1991).
- Leonard, B.P., 1981,
A survey of finite differences with upwinding for numerical modelling of incompressible convective diffusion equation, *Recent Advances in Numer Methods in Fluids*, Vol. 2, ed., C. Taylor, Pineridge Press, England , (1981).
- Liu, Y and Chen. Y, 1989,
Finding in using the program : a versatile two dimensional mesh generator with automatic band width reduction, *Comp. and Struct.*, Vol. 32, No. 2, pp. 145-149, (1989).
- Lo, Sh., 1991,
Automatic mesh generation and adaption by using contours , *Int. J. Num. Engg.*, Vol. 31, No. 4, pp. 689-708, (1991).
- Lohner, R., Morgan, K. and Zienkiewicz, 1986,
Adaptive grid refinement for the compressible Euler equations, in *Accuracy Estimates and Adaptive Refinements in Finite Element Computations*, (Eds. I. Babuska et al.) Wiley, New York, (1986).
- Lohner, R. and Morgan, 1987,
Improved adaptive refinement strategies for finite element aerodynamic computations, *AIAA Paper 87-055*, (1987).
- Lohner, R., 1987,
Adaptive finite element scheme for transient problems in *CFDComp. Meth. Appl. Mech. Engg.*, Vol. 61, No. 3, pp. 323-338, (1987)
- Ludford, G.S.S. and Peters, N., 1984,
Dynamics of flames and reactive systems, Bowen, J.D. et al., eds., Vol. 95, pp. 75, *Progress in Astronautics and Aeronautics*, AIAA, NY, 1984.
- Mack, A., Weber, H. J., and Roth P., 1992,
A moving grid method applied to one dimensional non stationary flame propagation, *Int. J. Num. Meth. Fluids.*, Vol. 13, pp. 869-882, (1992).
- Mao, K. M., and Sun, CT., 1991,
A refined global - local finite element analysis method, *Int. J. Num. Meth. Engg.*, Vol. 32. No. 1, pp. 29 - 44, (1991).
- Mastin, C. W. and Thompson, J. F. 1978,
Elliptic systems and numerical transformations, *J. Math.*

- Anal. Appl.*, Vol. 62, pp. 52-62, (1978).
- Mastin, C.W., 1988,
Fast interpolation schemes for moving grids, in Sengupta, et al. (eds.), *Numerical Grid Generation in Computational Fluid Mechanics*, Pineridge Press, Swansea, UK, pp. 63-74, (1988).
- Miller, R. and Miller, R. N., 1981a,
Moving finite elements. I, *SIAM. J. Num. Analy.*, Vol. 18, No. 6, pp. 1019-1032, (1981).
- Miller, R. and Miller, R. N., 1981b,
Moving finite elements. II, *SIAM. J. Num. Analy.*, Vol. 18, No. 6, pp. 1032-1057, (1981).
- Morgan, K., Peraire, J., Thareja, R. R. and Stewart, J. R., 1987,
An adaptive finite element scheme for the Euler and Navier-Stokes equations, Presented at the *AIAA 8th Computational Fluid Dynamics Conference*, Hawai, June, (1987).
- Morris, R. B., Tsuji, Y. and Carnevali, P., 1992,
Adaptive solution strategies for solving large systems of p-type finite element equations, *Int. J. Num. Meth. Engg.* Vol. 33, No.10, pp. 2059-2072, (1992).
- Nakahashi, K. and Deiwert, G. S., 1985,
A practical adaptive grid method for complex fluid flow problems, *Lecture notes in Physics*, 218, Springer Verlag, pp. 422-426, (1985).
- Nakumara, S., 1982,
Marching grid generation using parabolic partial equations, *Numerical Grid Generation*, 1st Ed, New York, North Holland, pp. 775-786, (1982).
- O' Brien, V., 1981,
Conformal mappings for internal viscous flow problems, *J. Comp. Phys.*, Vol. 44, pp. 220-226, (1981).
- Oden, J. T. and Reddy, J. N., 1976,
Mathematical Theory of Finite Elements, Academic Press, NY, (1976).
- Oden, J. T., Devloo, P. and Strouboulis, T., 1986a,
Adaptive finite element methods for the analysis of inviscid compressible flow. I. Fast refinement/unrefinement and moving mesh methods for unstructured meshes, *Comp. Meth. Appl. Mech. Engg.*, Vol. 59, No. 3, pp. 327-362, (1986).
- Oden, J. T. and Demkowicz, L., 1986a,
An adaptive characteristic Petrov - Galerkin finite element method for convection dominated linear and nonlinear parabolic problems in one space variable, *J. Comp. Phys.*, Vol. 67, pp. 188-213, (1986).

- Oden, J.T. and Demkowicz, L., 1986b,
Advances in adaptive improvements : A survey of adaptive finite element methods in computational mechanics, *State of the Art Survey in Computational Mechanics*, A Special Publication of The American Society of Mechanical Engineering, (1986).
- Oden, J.T., Demkowicz, L., Rachowicz, W. and Westermann, T. A., 1989,
Toward a universal h-p adaptive finite element strategy, Part 2. a posteriori error estimation ', *Comp. Meth. Appl. Mech. Engg.*, Vol. 77, No. 1-2, pp. 113-180, (1989).
- Oden, J. T. and Demkowicz, L., 1991,
h-p adaptive finite element methods in computational fluid dynamics, *Comp. Meth. Appl. Mech. Engg.*, Vol. 89, No. 1-3, pp. 11-40, (1991).
- Oran, Elaine S and Bores, J. P., 1981,
Detailed modelling of combustion systems, *Progress in Energy and Combustion Science*, Vol. 7, pp. 1-72, (1981).
- Patel, M.K., Markatos, N.C. and Cross, M., 1985,
A critical evaluation of seven discretization schemes for convection diffusion equations, *Int. J. Num. Meth. Fluids*, Vol. 5, pp. 225-244, (1985).
- Peters, N. and Williams, F. A, 1984
Dynaics of Flames and Reactive Systems. Bowen et al., *Progress in Astronautics and Aeronautics*, Vol. 95, pp. 37, AIAA, NY, (1984).
- Rachowicz, L., Oden, J. T., and Demkowicz, L., 1989,
Toward a universal h-p adaptive finite element strategy, Part 3. design of meshes, *Comp. Meth. Appl. Mech. Engg.*, Vol. 77, No. 12, pp. 181-212, (1989).
- Rai, M. M., and Anderson, D. A., 1981,
Grid evolution in time asymptotic problems, *J. Comp. Phys.*, Vol. 43, pp. 327-344, (1981).
- Rai, M. M. and Anderson, D. A., 1982,
Application of adaptive grids to fluid flow problems with asymptotic solutions, *AIAA. J.*, Vol. 20, pp. 496-502, (1982).
- Raithby, G.D., 1976
Skew upstream differencing schemes for problems involving fluid flow, *Comput. Meth. Appl. Mech. Engg.*, Vol. 9, pp. 153-164, (1976).
- Ramos, J. I., 1983,
Numerical studies of laminar flame propagation in spherical bombs, *AIAA. J.*, Vol. 21, No. 3, pp. 415-422, (1983).

- Ramos, J. I., 1985,
Development and application of an adaptive finite element to reaction diffusion equations, *Int. J. Num. Meth. Fluids*, Vol. 5, pp. 13-23, (1985).
- Ramos, J.I., 1987,
Numerical methods for one-dimensional reaction-diffusion equations arising in combustion theory, in T.C. Chawla (ed), *Annual Review of Numerical Fluid Mechanics and Heat Transfer*, (Hemisphere, New York). Vol. 1. pp. 150-261, (1987).
- Ramos, J.I. and Shif, T.I.P., 1987,
Numerical solution of reaction-diffusion equations by compact operators and modified equation methods, *Int. J. Num. Meth. Fluids*, Vol. 2, No. 4, pp. 337-351, (1987).
- Ramos, J.I., 1990a,
Finite element methods for one-dimensional flame propagation problems, in T.J. Chung (ed), *Numerical Modelling in Combustion* (Hemisphere, New york), (1990).
- Ramos, J. I., 1990b,
Finite element methods for one-dimensional combustion problems, *Int. J. Num. Meth. Fluids*, Vol. 11, No. 6, pp. 893-906, (1990).
- Ramos, J.I., 1991,
Adaptive and nonadaptive Hermitian operator methods for combustion phenomena, *Comp. Meth. Appl. Mech. Engg.*, Vol. 90, No. 1-3, pp. 609-630, (1991).
- Rao, V. C. V., Sundararajan, T. and Das, P. C., 1988,
A new approach to grid generation using finite element technique, in Sengupta, et al. (eds.), *Numerical Grid Generation in Computational Fluid Mechanics*, 88, Pineridge Press, Swansea, UK, pp. 157-166, (1988).
- Rheinboldt, W. C., 1981,
Adaptive mesh refinement processes for finite element solutions, *Int. J. Num. Meth. Engg.*, Vol. 17, No. 5, pp. 649-662, (1981).
- Rheinboldt, W. C., 1983,
Feedback systems and adaptivity for numerical computations, in I. Babuska, J. Chandra and J. E. Flaherty (eds.), *Adaptive Computational Methods for Partial Differential Equations* , SIAM, Philadelphia, (1983).
- Rivara, M.C., 1984,
Algorithms for references triangular grids suitable for adaptive and multigrid techniques, *Int. J. Num. Meth., Engg.*, Vol. 20, pp. 745-756, (1984).

- Saltzman, J. and Brackbill, K., 1982,
Applications and generalizations of variational methods for
generating adaptive meshes, in J. F. Thompson (ed),
Numerical Grid Generation, North-Holland, Amsterdam, pp.
865-884, (1982).
- Sampaio, P. A. B. DE., 1990,
Petrov-Galerkin/modified operator formulation for
convection-diffusion problems, *Int. J. Num. Meth. Engg.*, Vol.
30, pp. 331-347, (1990).
- Sang, H. and Hsieh S.S., 1991,
Applications of a self-adaptive algorithm to non-linear
finite element analysis, *Int. J. Num. Meth. Engg.*, Vol. 32,
No. 5, pp. 1057-1077, (1991).
- Sengupta, S., Hauser, J., Eiseman, P. R., and Thompson, J. F.,
(eds.) 1988,
Numerical grid generation in computational fluid mechanics
1988, *Second International Conference on Numerical Grid
Generation in Computational Fluid dynamics*, Miami, (1988)
Pineridge Press, Swansea, UK, (1988).
- Shepherd, M. S., Yerry, M. A. and Baehmann, P. L., 1986,
Automatic mesh generation allowing for efficient a priori and
posteriori mesh refinement, *Comp. Meth. Appl. Mech. and
Engg.*, Vol. 55, No. 1-2, pp. 161-180, (1986).
- Smith, R.E., 1983,
Three dimensional algebraic grid generation, *AIAA Paper*
83-1904, Danvers, Mass., (1983).
- Smith, R. and Eriksson, L. E., 1986,
Algebraic grid generation, *Comp. Meth. Appl. Mech. Engg.*,
Vol. 64, Nos. 1-3, pp. 285-300, (1986).
- Smooke, M. D., Miller, J. A. and Kee, R. J., 1985,
Solution of premixed and counterflow diffusion flame problems
by adaptive boundary value methods, Report ME-102-85, Yale
university, (1985).
- Smooke, M. D. and Koszykowski, M. L., 1986,
Fully adaptive solutions of one dimensional mixed
initial-boundary value problems with applications to unstable
problems combustion, *SIAM. J. Sci. Stat. Comput.*, Vol. 7, pp.
301, (1986).
- Smooke, M. D., 1986,
On the use of adaptive grids in premixed combustion, *AICHE*.
32, pp. 1233-1241, (1986).
- Sorenson. R. L. and Steger, J. L., 1980,
Numerical generation of two dimensional grids by the use of
poisson equations with grid control at boundaries, *Numerical*

- Grid Generation*, pp. 11-40, (1991).
- Steger, J. L. and Sorenson. R. L., 1979,
Automatic mesh point clustering near a boundary in grid
generation for elliptic partial differential equations, *J.
Comp. Phys.*, Vol. 33, pp. 405-416, (1979).
- Suzuki. M., 1991,
Surface grid generation based on unstructured grids, *AIAA.
J.*, Vol. 29, No. 12, pp. 2262-2264, (1991).
- Szabo, B. A., 1986,
Mesh design for the p version of the finite element method,
Comp. Meth. Appl. Mech. Engg., Vol. 55, No. 1-2, pp. 181-197,
(1986)
- Szymczak, W. G., and Babuska, I., 1984,
Adaptivity and error estimation for the finite element method
applied to convection diffusion problems , *SIAM. J. Num.
Analy.*, Vol. 21, No 5, pp. 910-954, (1984).
- Talbert, J. A. and Parkinson, A. R., 1990,
Development of an automatic two dimensional finite element
mesh generator using quadrilateral elements and beizier curve
boundary definition, *Int. J. Num. Meth. Engg.*, Vol. 29, No. 7,
pp. 1551-1568, (1990).
- Taylor, C. and Hughes, T. G., 1981,
*A Finite Element Programming of the Navier-Stokes
Equation*, Pineridge Press, Swansea, UK, (1981).
- Thacker, W. C., 1980,
A brief review of techniques for generating irregular
computational grids, *Int. J. Num. Meth. Engg.*, Vol. 15, No. 9,
pp. 1335-1341, (1980).
- Tezduyar, T. E., and Park, Y. J., 1986,
Discontinuity-capturing finite element formulations for
non-linear convection-diffusion reaction equations, *Comp.
Meth. Appl. Mech. Engg.*, Vol. 59, No. 3, pp. 307-325, (1986).
- Thompson, J. F., Thames, F. C. and Mastin, C. W., 1977,
TOMCAT- A code for numerical generation of boundary-fitted
curvilinear coordinate systems on fields containing any
number of arbitrary two-dimensional bodies, *J. Comp. Phys.*,
Vol. 24, No. 3, pp. 245-273, (1977).
- Thompson, J. F., 1982a,
Numerical Grid Generation, North Holland, Amsterdam, (1982).
- Thompson, J. F, 1982b,
Elliptical grid generation, in J.F. Thompson (ed), *Numerical
Grid Generation*, North Holland, pp. 79-105, (1982).

- Thompson, J. F., Warsi, Z. U. A. and Mastin, C. W., 1982,
Boundary-fitted coordinate systems for numerical solution of
partial differential equations - A Review, *J. Comp. Phys.*,
Vol. 47, pp. 1-108, (1982).
- Thompson, J. F., 1984,
Grid generation techniques in computational fluid dynamics,
AIAA. J., Vol. 22, No. 11, pp. 1505-1523, (1984).
- Thompson, J.F., Warsi, Z.U.A. and Mastin, C. W., 1985a,
Numerical Grid Generation Foundations and Applications, North
Holland, Amsterdam, (1985).
- Thompson, J.F., Warsi, Z.U.A. and Mastin, C. W., 1985b,
Adaptive grids, in Thompson et al. (eds.), *Numerical Grid
Generation Foundations and Applications*, North Holland,
Amsterdam, pp. 367-419, (1985).
- Thompson, J.F., Warsi, Z.U.A. and Mastin, C. W., 1985c,
Elliptic generation systems, in Thompson et al.
(eds.), *Numerical Grid Generation and Applications*, North
Holland, Amsterdam, pp. 188-263, (1985).
- Thompson, J. F., 1985,
A survey of dynamically adaptive grids in the numerical
solution of partial differential equations, *Appl. Num. Math.*,
Vol. 1, pp. 3-27, (1985).
- Turkel, E., 1983,
Progress in computational physics, *Computers and Fluids*, Vol.
11, No. 2, pp. 122-144, (1983).
- Williams, F.A, 1985,
Combustion theory, second edition, Benjamin/Cummings, Menlo
Park, CA, (1985)
- Yerry, M.A., and Shepherd, M.S., 1984
Automatic three-dimensional mesh generation by the modified
-octree technique, *Int. J. Num. Meth. Engg.*, Vol. 20, pp.
1965-1990, (1984).
- Yerry, M. A. and Shepherd, M. S., 1988,
A modified quadtree approach to finite element grid
generation, *IEEE Graphics J.*, pp. 31-47, (1988).
- Yuen, M. M. F., Tan, ST., and Hung, KY., 1991,
A hierarchical approach to automatic finite element mesh
generation, *Int. J. Num. Meth. Engg.*, Vol. 32, No. 3, pp.
501-526, (1991).
- Zhu, J., Rodi, W. and Schoenung, B., 1988,
Algebraic generation of smooth grids, in Sengupta, et al.
(eds.), *Numerical Grid Generation in Computational Fluid
Mechanics*, Mechanics 88, Pineridge Press, Swansea, UK, pp.

217-266, (1988).

Zhu, J. Z., Zienkiewicz, O. C., Hinton, E. and Wu. J., 1991,
A new approach to the development of quadrilateral mesh
generation, *Int. J. Num. Meth. Engg.*, Vol. 32, No. 4, pp.
849-866, (1991).

Zhu, J. Z., and Zienkiewicz, O. C., 1991,
Adaptivity and mesh generation, *Int. J. Num. Meth. Engg.*,
Vol. 32, No. 4, pp. 783-810, (1991)

Zienkiewicz., O.C. and Phillips, D.V., 1971
An automatic mesh generation scheme for plane and curved
surfaces by isoparametric co-ordinates, *Int. J. Num. Meth.
Engg.*, Vol. 3, pp. 519-528, (1971).

Zienkiewicz, O. C. and Heinrich, J. C., 1978,
The finite element method and convection problem in fluid
mechanics, in R. H. Gallagher et al. (eds.), *Finite Elements
in Fluids*, Vol. 3., Wiley, (1978).

Zienkiewicz, O.C., de S. R. Gago, J. P. and Kelly, D. W., 1983,
The hierarchical concept in finite element analysis, *Comp.
and Struct.*, Vol. 16, No. 1-4, pp. 53-65, (1983).

Zienkiewicz, O. C., Zhu, J. Z., and N. G. Gong, 1989,
Effective and practical h-p version adaptive analysis
procedures for the finite element method, *Int. J. Num. Meth
Engg.*, Vol. 28, No. 4, pp. 879-891, (1989).

Zienkiewicz, O.C. and Heinrich, J.C., 1978,
The finite element method and conduction convection problems
in fluid mechanics, in *Finite elements in fluids*, Gallagher,
Zienkiewicz, Oden Morandi, Cecchi and Taylor, eds., John
Wiley, London, pp. 1-21, (1978).

ADDENDUM :

On page 173 the text is modified to include the following paragraph at the end of section 4.8.1. of Chapter 4.

On the other hand, if the front speed is not a constant with respect to time, the success of the scheme depends on how accurately the front speed can be predicted. If this prediction is not accurate, it is quite possible that errors may accumulate, due to the discrepancy between the velocities of the front and the grid; in course of time, the numerical simulation may become unstable because of error accumulation.